



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF CONTROL AND INSTRUMENTATION

# ALGORITMY ODBUZOVÁNÍ PŘI ŘÍZENÍ ASYNCHRONNÍHO MOTORU

AC INDUCTION MACHINE FIELD WEAKENING ALGORITHMS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

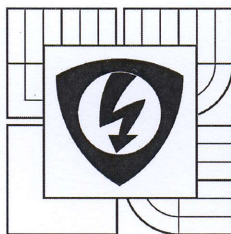
ZBYNĚK MYNÁŘ

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. PAVEL VÁCLAVEK, Ph.D.

BRNO 2012



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav automatizace a měřicí techniky

# Bakalářská práce

bakalářský studijní obor  
Automatizační a měřicí technika

**Student:** Zbyněk Mynář

**Ročník:** 3

**ID:** 125558

**Akademický rok:** 2011/12

**NÁZEV TÉMATU:**

## Algoritmy odbuzování při řízení asynchronního motoru

### POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s modelováním a řízením asynchronního motoru. Prostudujte existující a případně navrhnete nové algoritmy odbuzování pro řízení ve vysokých otáčkách. Vybrané algoritmy implementujte v prostředí Matlab-Simulink a ověřte jejich vlastnosti. Algoritmy následně připravte pro realizaci na procesorech s pevnou řádovou čárkou, implementujte je na procesoru řady Freescale 56F800E a ověřte simulaci metodou Processor-in-the-loop.

### DOPORUČENÁ LITERATURA:

[1] Caha, Z.; Černý, M. : Elektrické pohony, Praha, SNTL 1990.

další dle pokynů vedoucího

**Termín zadání:** 6.2.2012

**Termín odevzdání:** 28.5.2012

**Vedoucí práce:** doc. Ing. Pavel Václavek, Ph.D.

**Konzultanti bakalářské práce:**

doc. Ing. Václav Jirsík, CSc.

předseda oborové rady



### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

## **Abstrakt**

V současnosti jsou díky pokroku ve výpočetní a výkonové technice asynchronní motory nasazovány tam, kde bylo dříve možno použít pouze elektromotory jiných typů. Ty obvykle postrádají spolehlivost a nízké pořizovací náklady asynchronních motorů. Nejlepší výsledky při řízení asynchronních motorů vykazuje vektorové řízení, jemuž je věnována část práce. Pokud motor chceme provozovat v otáčkách pro něž nedostačuje ani maximální velikost statorového napětí, nabízí se možnost motor odbudit, čili snížit magnetický tok a tím těchto provozních hodnot dosáhnout. Algoritmů odbuzování existuje celá řada. Právě srovnání několika nejčastěji používaných algoritmů je věnována tato práce. Jako simulačního nástroje bylo použito prostředí MATLAB-Simulink. Další část práce je věnována implementaci těchto algoritmů do procesoru řady 56F800E užívajícího výpočty v pevné řádové čárce.

## **Klíčová slova**

Asynchronní motor, odbuzování, vektorové řízení, matematický model

## **Abstract**

Nowadays, thanks to advances in information and power technology induction motors are deployed there, where previously only other types of electric motors were possible to use. Those typically lack the reliability and low cost of induction motors. Best results in control of induction motors are achieved when vector control is used, to which part of the work is paid. If induction motor reaches the speed when it is no longer possible to continue to increase stator voltage, there is possibility of field weakening or reduction of magnetic flux and thus achieve higher speed. A comparison of several algorithms is devoted to this work. As a simulation tool MATLAB-Simulink was used. Another part is devoted to implementation of these algorithms to the processor of 56F800E series which uses calculations in fixed point.

## **Keywords**

Induction motor, field weakening, vector control, mathematical model

### **Bibliografická citace:**

MYNÁŘ, Zbyněk. *Algoritmy odbuzování při řízení asynchronního motoru*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Ústav automatizace a měřicí techniky, 2012. 71 s. Vedoucí bakalářské práce byl doc. Ing. Pavel Václavek, Ph.D.

## Prohlášení

„Prohlašuji, že svou bakalářskou práci na téma Algoritmy odbuzování při řízení asynchronního motoru jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.“

V Brně dne: **23. května 2012**

.....  
podpis autora

## **Poděkování**

Děkuji vedoucímu bakalářské práce doc. Ing. Pavlovi Václavkovi, Ph.D, za jeho ochotu, trpělivost a cenné odborné rady, jež mi poskytl při vypracování této bakalářské práce

V Brně dne: **23. května 2012**

.....  
podpis autora

# OBSAH

<b>OBSAH .....</b>	<b>8</b>
<b>SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK .....</b>	<b>10</b>
<b>SEZNAM OBRÁZKŮ .....</b>	<b>13</b>
<b>SEZNAM TABULEK.....</b>	<b>14</b>
<b>ÚVOD .....</b>	<b>15</b>
<b>1 MATEMATICKÝ MODEL ASYNCHRONNÍHO MOTORU .....</b>	<b>16</b>
1.1 Clarkové transformace .....	16
1.2 Parkova transformace .....	17
1.3 Odvození matematického modelu .....	19
<b>2 VEKTOROVÉ ŘÍZENÍ.....</b>	<b>22</b>
2.1 Odvození vektorového řízení .....	22
2.2 Odvození přenosů nutných pro návrh regulátorů .....	23
2.2.1 Proudový přenos .....	24
2.2.2 Přenos otáček a magnetického toku $\Psi_{2d}$ .....	25
2.3 Způsob návrhu regulátorů .....	26
2.4 Výpočet magnetického toku $\Psi_{2d}$ a jeho úhlové rychlosti.....	27
2.5 Výsledná struktura vektorového řízení.....	28
<b>3 VYBRANÉ ALGORITMY ODBUZOVÁNÍ .....</b>	<b>29</b>
3.1 Algoritmus $1/\omega_f$ .....	29
3.2 Odbuzování s PI regulátorem .....	31
3.3 Odbuzování na maximální moment.....	31
3.4 Algebraické odbuzování na maximální moment .....	35
<b>4 IMPLEMENTACE ŘÍDICÍ ČÁSTI ALGORITMU DO DSC 56F8346 .....</b>	<b>38</b>
4.1 Metoda Processor-in-the-loop .....	38
4.2 Normalizace rovnic vektorového řízení pro výpočty v pevné řádové čárce.....	40
4.2.1 Transformace souřadnicových systémů .....	41
4.2.2 Výpočet rotorového magnetického toku .....	42
4.2.3 Výpočet úhlové rychlosti magnetického toku .....	44
4.2.4 Normalizace diskrétního PI a PID regulátoru.....	44
4.3 Úprava vybraných algoritmů odbuzování pro výpočty v pevné řádové čárce.....	48
4.3.1 Algoritmus $1/\omega_f$ .....	48
4.3.2 Odbuzování s PI regulátorem .....	48
4.3.3 Odbuzování na maximální moment .....	49



4.3.4	<i>Algebraické odbuzování na maximální moment.....</i>	49
4.4	Výsledný algoritmus .....	51
<b>5</b>	<b>SIMULACE ALGORITMŮ ODBUZOVÁNÍ .....</b>	<b>53</b>
5.1	Simulace motoru s dominantním prvním regionem .....	54
5.2	Simulace pro motor s dosažitelným druhým regionem .....	57
5.3	Srovnání algoritmů odbuzování podle doby výpočtu .....	61
<b>ZÁVĚR .....</b>		<b>63</b>
<b>LITERATURA.....</b>		<b>65</b>
<b>SEZNAM PŘÍLOH.....</b>		<b>67</b>

# SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

$e$	Odchylka $e=w-y$	
$i_1$	Proud statorovým vinutím	[A]
$i_2$	Proud rotorovým vinutím	[A]
$ITAE$	Hodnota $ITAE$ kritéria, pro úhlovou rychlost $\omega$	[rad]
$j$	Imaginární jednotka	[-]
$J$	Moment setrvačnosti	[kgm <sup>2</sup> ]
$k$	Pořadí vzorku	[-]
$k_x$	Volitelný parametr nižší, blízký jedné u algoritmu odbuzování $x$	[-]
$k_{FWA}$	Maximální doba výpočtu algoritmu odbuzování v cyklech	[-]
$k_{VOC}$	Maximální doba výpočtu algoritmu vektorového řízení v cyklech	[-]
$K$	Zesílení regulátoru	[-]
$L_1$	Vlastní indukčnost fáze statorového vinutí	[H]
$L_2$	Vlastní indukčnost fáze rotorového vinutí	[H]
$L_h$	Vzájemná indukčnost fáze rotoru a statoru při uvažování vlivu všech fází rotoru	[H]
$m$	Točivý moment motoru	[Nm]
$m_z$	Zátěžný moment	[Nm]
$M_v$	Měřítka obecné veličiny $v$	[-]
$N$	Konstanta filtrace derivační složky	[-]
$N_v$	Norma obecné veličiny $v$	[jednotka $v$ ]
$P_p$	Počet pólových párů motoru	[-]
$R_1$	Odpor fáze statorového vinutí	[Ω]
$R_2$	Odpor fáze rotorového vinutí	[Ω]
$s$	Skluz motoru	[-]
$t$	Čas	[s]
$T_d$	Derivační časová konstanta	[s]
$T_i$	Integrační časová konstanta	[s]
$T_s$	Vzorkovací perioda	[s]
$T_{s2}$	Vzorkovací perioda algoritmu odbuzování, kdy $T_{s2}=2^n \cdot T_s$	[s]
$T_{sim}$	Délka časového kroku výpočtu simulace	[s]
$T_t$	Konstanta zpětného výpočtu (anti-windup)	[s]
$u_1$	Statorové napětí	[V]
$u_2$	Rotorové napětí	[V]

$U_m$	Amplituda statorového napětí	[V]
$w$	Žádaná hodnota	
$x$	Akční zásah	
$y$	Regulovaná veličina	
$\delta_{FWA}$	Relativní srovnání dob výpočtu $k_{FWA}$ a $k_{VOC}$	[%]
$\delta_{ITAE}$	Relativní srovnání hodnoty $ITAE$ , s nejlepším výsledkem	[%]
$\theta_f$	Úhel natočení magnetického toku $\Psi_2$ vůči souřadnému systému statoru	[rad]
$\theta_K$	Úhel natočení souřadného systému K vůči souřadnému systému statoru	[rad]
$\theta$	Úhel natočení veličiny vůči souřadnému systému statoru	[rad]
$\sigma$	Činitel celkového rozptylu	[-]
$\tau_{1d}$	Časová konstanta v pólu přenosu proudu $i_{1d}$	[s]
$\tau_{1q}$	Časová konstanta v pólu přenosu proudu $i_{1q}$	[s]
$\tau_1$	Časová konstanta statoru při rotorovém vinutí nakrátko	[s]
$\tau_2$	Časová konstanta rotoru při statorovém vinutí nakrátko	[s]
$\Psi_1$	Spřažený statorový magnetický tok	[Wb]
$\Psi_2$	Spřažený rotorový magnetický tok	[Wb]
$\omega_e$	Synchronní úhlová rychlost vektoru napětí napájecí sítě	[rad/s]
$\omega_f$	Úhlová rychlost rotorového magnetického toku $\Psi_{2d}$	[rad/s]
$\omega_K$	Úhlová rychlost rotujícího souřadného systému	[rad/s]
$\omega_m$	Úhlová rychlost rotoru	[rad/s]
$\omega_s$	Úhlová rychlost vektoru statorového napětí	[rad/s]

#### Seznam zkratk:

ACIM	AC Induction Motor (asynchronní motor)
DSC	Digital Signal Controller (digitální signálový kontrolér)
ITAE	Integral of Time Absolute Error
PWM	Pulse Width Modulation (pulsní šířková modulace)
SFIO	S-Function Input Output
VOC	Vector Oriented Control (vektorově orientované řízení)

**Význam horních indexů:**

<i>max</i>	Limitace veličiny
<i>S</i>	Statorový souřadný systém
<i>ref</i>	Reference, neboli žádaná hodnota
<i>R</i>	Rotorový souřadný systém
<i>K</i>	Rotující souřadní systém

**Význam dolních indexů:**

1	Statorová veličina
2	Rotorová veličina
<i>n</i>	Bezrozměrná normalizovaná veličina

# SEZNAM OBRÁZKŮ

Obr. 1: Princip Clarkové transformace pro statorové napětí.....	16
Obr. 2: Princip Parkovy transformace pro statorové napětí [2] .....	19
Obr. 3: Matematický model asynchronního motoru [1].....	21
Obr. 4: Schéma algoritmu odbuzování $1/\omega_f$ v prostředí MATLAB-Simulink .....	30
Obr. 5: Schéma algoritmu odbuzování s PI regulátorem v prostředí MATLAB-Simulink .....	31
Obr. 6: Napěťové omezení v napěťových souřadnicích [9] .....	32
Obr. 7: Optimální napěťový vektor, při odbuzování v prvním regionu [9] .....	33
Obr. 8: Optimální napěťový vektor, při odbuzování v druhém regionu [9].....	34
Obr. 9: Realizace algoritmu odbuzování na maximální moment v MATLAB-Simulink .....	34
Obr. 10: Napěťové a proudové omezení v proudových souřadnicích.....	35
Obr. 11: Realizace algoritmu odbuzování na maximální moment algebraicky v prostředí MATLAB-Simulink .....	37
Obr. 12: Znáznornění metody Processor-in-the-loop pomocí SFIO Toolboxu.....	39
Obr. 13: Výsledné schéma simulace vektorového řízení s využitím metody Processor-in-the-loop v MATLAB-Simulink.....	39
Obr. 14: Přetečení při počítání v pevné řádové čárce .....	40
Obr. 15: Výpočet magnetického toku z modelu v prostředí MATLAB-Simulink.....	42
Obr. 16: Realizace použitého paralelního PI regulátoru .....	46
Obr. 17: Realizace použitého paralelního PID regulátoru .....	47
Obr. 18: Jednoduchý časový diagram řazení výpočtů pro $n=4$ .....	51
Obr. 19: Časová závislost magnetického toku u motoru s dominantním prvním regionem .....	55
Obr. 20: Časová závislost momentu u motoru s dominantním prvním regionem.....	55
Obr. 21: Časová závislost úhlové rychlosti $\omega_f$ u motoru s dominantním prvním regionem.....	56
Obr. 22: Časová závislost mechanické úhlové rychlosti u motoru s dominantním prvním regionem.....	56
Obr. 23: Časová závislost magnetického toku u simulace motoru s dosažitelným druhým regionem.....	58
Obr. 24: Časová závislost momentu u simulace motoru s dosažitelným druhým regionem.....	59
Obr. 25: Časová závislost úhlové rychlosti $\omega_f$ u simulace motoru s dosažitelným druhým regionem.....	59
Obr. 26: Časová závislost úhlové rychlosti rotoru u motoru s dosažitelným druhým regionem	60
Obr. 27: Srovnání algoritmů podle výpočetní doby .....	61

# SEZNAM TABULEK

Tab. 1: Vstupní parametry simulace motoru s dominantním prvním regionem odbuzování .....	54
Tab. 2: Hodnoty ITAE kritéria dosažených algoritmy odbuzování při simulaci motoru s dominantním prvním regionem .....	57
Tab. 3: Vstupní parametry simulace motoru s dosažitelným druhým regionem.....	58
Tab. 4: Hodnoty ITAE kritéria dosažených algoritmy odbuzování při simulaci motoru s dosažitelným druhým regionem.....	60
Tab. 5: Výpočetní náročnost algoritmů odbuzování, vzhledem k celkové době výpočtu vektorového řízení .....	62

# ÚVOD

Asynchronní motory jsou nejčastěji používanými elektrickými pohony. Díky jejich jednoduché konstrukci, nevyžadující drahé materiály, jsou levné a především spolehlivé. Ve většině případů jsou napájeny přímo z rozvodné sítě, bez možnosti regulace otáček. Na rozdíl od synchronního elektromotoru je, z hlediska složitosti řízení, nevýhodou absence pevné vazby mezi rotorovým magnetickým tokem a fyzickým vinutím rotoru, přičemž je k dispozici pouze jeden řídicí vstup. K indukci napětí na rotorovém vinutí dochází až při rozdílu úhlové frekvence statorového magnetického toku a otáček rotoru (odtud název asynchronní motor). Krom zastaralých způsobů řízení otáček, jako změnou rotorového odporu u kroužkových motorů či změnou skluzu, se dnes nejčastěji využívá frekvenčního nebo vektorového řízení, kterému je věnována část této práce.

U některých aplikací se můžeme setkat s potřebou provozovat motor v otáčkách, jež nejsme, z důvodů napěťového omezení napájení, schopni normálně dosáhnout. Způsobem, jak těchto otáček docílit, je motor odbudit, tedy zmenšit magnetický tok, který je při nízkých otáčkách držen na maximální hodnotě, pro jakou je motor konstruován, tedy hodnotě při napájení z rozvodné sítě. Algoritmů odbuzování byla vyvinuta celá řada. V této práci bude věnována pozornost čtyřem vybraným algoritmům, jejichž hlavní vlastnosti budou srovnány pomocí simulace v prostředí MATLAB-Simulink verze R2008b.

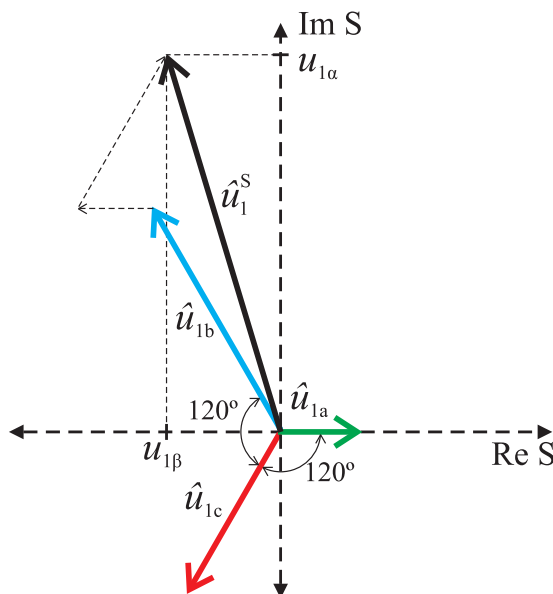
Využití výpočetně náročných algoritmů, jako je vektorové řízení, bylo umožněno až s rozšířením digitálních signálových kontrolérů (DSC), mezi něž patří i procesory řady 56F800E firmy Freescale. Implementaci řídicí části algoritmů vektorového řízení a algoritmů odbuzování pro tyto procesory je věnována další část práce. Díky metodě Processor-in-the-loop bylo možno implementované algoritmy ověřit simulací v prostředí MATLAB-Simulink.

# 1 MATEMATICKÝ MODEL ASYNCHRONNÍHO MOTORU

Ve srovnání s jinými elektrickými motory je matematický model asynchronního motoru relativně složitý. Při jeho vytváření je nutno přijmout několik zjednodušujících předpokladů. Zaprvé předpokládáme trojfázovou symetrii motoru, ve smyslu rozložení vinutí rotoru a statoru, jejich indukčností a odporů. Vinutí by měla zajistit harmonické rozložení magnetického pole ve vzduchové mezeře. Nelinearita magnetického obvodu a ztráty v železe jsou zanedbány. Dále hodnoty odporů a indukčností jsou konstantní a rotorové veličiny jsou přepočteny na stator. Neméně důležitým předpokladem je pak napájení trojfázovým symetrickým napětím.

## 1.1 Clarkové transformace

Aby byl získán co nejjednodušší matematický popis motoru, je trojfázový, tedy přirozený, souřadný systém  $a, b, c$  transformován na dvoufázový souřadný systém  $\alpha, \beta$ . To je provedeno zobrazením vektorů fází veličiny do Gaussovy roviny a jejich následným sečtením. Znázornění pro statorové napětí ve statorových souřadnicích  $S$  je na obrázku Obr. 1. Tato transformace je nazvána podle MSc. Edith Clarkové.



Obr. 1: Princip Clarkové transformace pro statorové napětí



Pro jednotlivá fázová napětí v harmonickém ustáleném stavu je možno psát

$$\begin{aligned} u_{1a} &= U_m \cos \omega_e t = U_m \frac{e^{j\omega_e t} + e^{-j\omega_e t}}{2} \\ u_{1b} &= U_m \cos\left(\omega_e t - \frac{2\pi}{3}\right) = U_m \frac{e^{j\omega_e t} e^{-j\frac{2\pi}{3}} + e^{-j\omega_e t} e^{j\frac{2\pi}{3}}}{2} \\ u_{1c} &= U_m \cos\left(\omega_e t + \frac{2\pi}{3}\right) = U_m \frac{e^{j\omega_e t} e^{j\frac{2\pi}{3}} + e^{-j\omega_e t} e^{-j\frac{2\pi}{3}}}{2}, \end{aligned} \quad (1.1.1)$$

kde  $U_m$  je amplituda statorového napětí a  $\omega_e$  jeho frekvence. Sečtením je získán výsledný komplexor

$$\hat{U}_1^S = K \frac{3U_m}{2} e^{j\omega_e t} = u_{1\alpha} + ju_{1\beta}, \quad (1.1.2)$$

kde  $K$  je volitelná konstanta [2]. Takovéto matematické operace by však byly pro mikroprocesor zbytečně složité. Výpočty lze zjednodušit na operace sčítání a násobení. Například pro statorové napětí platí při volbě  $K=2/3$  vztahy [2], [3], [4]

$$\begin{aligned} u_{1\alpha} &= u_{1a} \\ u_{1\beta} &= \frac{1}{\sqrt{3}} u_{1a} + \frac{2}{\sqrt{3}} u_{1b}. \end{aligned} \quad (1.1.3)$$

Díky těmto zjednodušeným vztahům je potřeba měřit pouze dva fázové proudy. Zpětnou transformaci Clarkové lze provést [2], [3], [4]

$$\begin{aligned} u_{1a} &= u_{1\alpha} \\ u_{1b} &= -\frac{1}{2} u_{1\alpha} + \frac{\sqrt{3}}{2} u_{1\beta} \\ u_{1c} &= -\frac{1}{2} u_{1\alpha} - \frac{\sqrt{3}}{2} u_{1\beta}. \end{aligned} \quad (1.1.4)$$

## 1.2 Parkova transformace

Další transformace souřadnic, jež velmi usnadní matematický popis asynchronního motoru se nazývá Parkova. Spočívá v roztočení souřadného systému  $S$  (souřadnice  $\alpha, \beta$ ), úhlovou rychlostí  $\omega_K$ , čímž je získán rotující souřadný systém  $K$  (souřadnice  $d, q$ ). Transformace je provedena vynásobením veličiny komplexorem s jednotkovou

amplitudou [2]. Princip je zobrazen na obrázku Obr. 2. Například pro statorová napětí platí

$$\hat{U}_1^K = U_m e^{j(\theta - \theta_K)} = \hat{U}_1^S e^{-j\theta_K} = u_{1d}^K + ju_{1q}^K, \quad (1.2.1)$$

kde  $U_m$  je amplituda statorového napětí,  $\theta$  je úhel natočení napětí vůči statorovému souřadnému systému a  $\theta_K$  je úhel natočení souřadného systému  $K$  vůči souřadnému systému  $S$ . V této formě je převod pro mikroprocesor opět složitou operací. Jednoduchou úpravou jsou získány rovnice ve složkovém tvaru. Pro přímou Parkovu transformaci lze psát

$$\begin{aligned} u_{1d}^K &= u_{1d}^S \cos \theta_K + u_{1q}^S \sin \theta_K \\ u_{1q}^K &= -u_{1d}^S \sin \theta_K + u_{1q}^S \cos \theta_K \end{aligned} \quad (1.2.2)$$

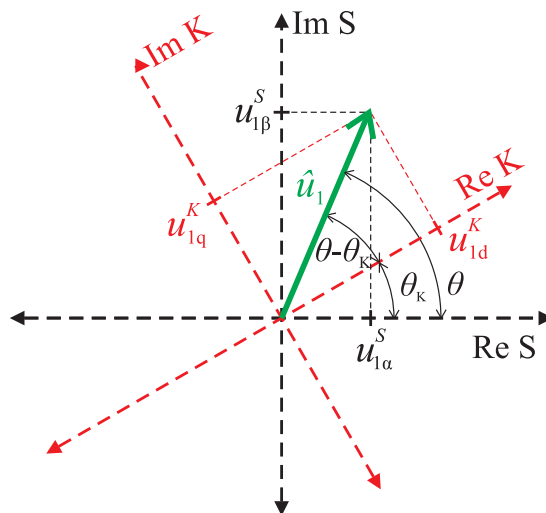
a pro zpětnou Parkovu transformaci [2], [3]

$$\begin{aligned} u_{1d}^S &= u_{1d}^K \cos \theta_K - u_{1q}^K \sin \theta_K \\ u_{1q}^S &= u_{1d}^K \sin \theta_K + u_{1q}^K \cos \theta_K. \end{aligned} \quad (1.2.3)$$

Použití na výpočet náročných goniometrických funkcí lze obejít přímým výpočtem  $\sin \theta_K$  a  $\cos \theta_K$ . Pokud bude platit  $\Psi_{2q}=0$ , pak při výpočtu rotorového magnetického toku platí [4]

$$|\Psi_2| = \sqrt{\Psi_{2\alpha}^2 + \Psi_{2\beta}^2} \quad (1.2.4)$$

$$\begin{aligned} \sin \theta_K &= \frac{\Psi_{2\beta}}{|\Psi_2|} \\ \cos \theta_K &= \frac{\Psi_{2\alpha}}{|\Psi_2|}. \end{aligned} \quad (1.2.5)$$



Obr. 2: Princip Parkovy transformace pro satorové napětí [2]

### 1.3 Odvození matematického modelu

Pro všechny veličiny provedeme Clarkové a Parkovu transformaci [1]. Pro satorové napětí ve satorovém souřadnicovém systému  $S$  je možno psát

$$\hat{u}_1^S = R_1 \hat{i}_1^S + \frac{d}{dt} \hat{\Psi}_1^S, \quad (1.3.1)$$

kde  $\hat{\Psi}_1^S$  je spřažený magnetický tok satoru. Pro rotorové napětí v rotorovém souřadném systému  $R$  platí

$$\hat{u}_2^R = R_2 \hat{i}_2^R + \frac{d}{dt} \hat{\Psi}_2^R = 0, \quad (1.3.2)$$

kde  $\hat{\Psi}_2^R$  je spřažený magnetický tok rotoru. Spřažené magnetické toky jsou dány vztahy

$$\begin{aligned} \hat{\Psi}_1^S &= L_1 \hat{i}_1^S + L_h e^{j\theta} \hat{i}_2^R \\ \hat{\Psi}_2^R &= L_2 \hat{i}_2^R + L_h e^{-j\theta} \hat{i}_1^S, \end{aligned} \quad (1.3.3)$$

kde  $L_1$  a  $L_2$  jsou vlastní indukčnosti satoru a rotoru,  $L_h$  je vzájemná indukčnost při nulovém  $\theta$ , což je úhel natočení mezi  $R$  a  $S$  souřadnými systémy. Nyní budou obě napěťové rovnice převedeny do společného rotujícího souřadného systému  $K$ , což je provedeno vynásobením obou stran rovnice (1.3.1) komplexorem  $e^{j\vartheta_K}$ , popřípadě  $e^{j(\vartheta_K - \vartheta)}$  pro rotorovou rovnici (1.3.2). To samé platí pro rovnice (1.3.3). Rovněž je zvolena úhlová rychlost, jakou se bude souřadný systém  $K$  otáčet. Logickou volbou je úhlová rychlost satorového napětí  $\omega_e$ , neboť bývá jako vstupní veličina známa. Jak se

později ukáže, vztah pro výsledný moment je na této volbě nezávislý. Pouze výpočet je více či méně složitý. Od tohoto místa budou dále všechny veličiny uváděny ve společném souřadném systému  $K$ , proto již nebude horní index uváděn. Výsledkem je soustava rovnic

$$\hat{u}_1 = R_1 \hat{i}_1 + \frac{d}{dt} \hat{\Psi}_1 + j\omega_e \hat{\Psi}_1 \quad (1.3.4)$$

$$\hat{u}_2 = R_2 \hat{i}_2 + \frac{d}{dt} \hat{\Psi}_2 + j(\omega_e - \omega_m) \hat{\Psi}_2 = 0$$

$$\begin{aligned} \hat{\Psi}_1 &= L_1 \hat{i}_1 + L_h \hat{i}_2 \\ \hat{\Psi}_2 &= L_2 \hat{i}_2 + L_h \hat{i}_1, \end{aligned} \quad (1.3.5)$$

kde  $\omega_m$  je úhlová rychlost rotoru a platí  $\omega_e - \omega_m = s\omega_e$ . Nyní budou z rovnic (1.3.5) vyjádřeny proudy. Zavedením činitele celkového rozptylu

$$\sigma = 1 - \frac{L_h^2}{L_1 L_2}, \quad (1.3.6)$$

jsou získány vztahy

$$\begin{aligned} \hat{i}_1 &= \frac{1}{\sigma L_1} \hat{\Psi}_1 - \frac{1-\sigma}{\sigma L_h} \hat{\Psi}_2 \\ \hat{i}_2 &= \frac{1}{\sigma L_2} \hat{\Psi}_2 - \frac{1-\sigma}{\sigma L_h} \hat{\Psi}_1. \end{aligned} \quad (1.3.7)$$

Následně jsou proudové rovnice (1.3.7) dosazeny do napět'ových rovnic (1.3.4) a vyjádřeny sprážené magnetické toky. Výsledné stavové rovnice jsou

$$\begin{aligned} \frac{d\hat{\Psi}_1}{dt} &= \hat{u}_1 - \left( \frac{R_1}{\sigma L_1} + j\omega_e \right) \hat{\Psi}_1 + \frac{R_1}{\sigma L_1} \frac{L_h}{L_2} \hat{\Psi}_2 = \\ &= \hat{u}_1 - j\omega_e \hat{\Psi}_1 + \frac{1}{\tau_1} \left( \hat{\Psi}_1 - \frac{L_h}{L_2} \hat{\Psi}_2 \right) \\ \frac{d\hat{\Psi}_2}{dt} &= - \left[ \frac{R_2}{\sigma L_2} + j(\omega_e - \omega_m) \right] \hat{\Psi}_2 + \frac{R_2}{\sigma L_2} \frac{L_h}{L_1} \hat{\Psi}_1 = \\ &= -j(\omega_e - \omega_m) \hat{\Psi}_2 + \frac{1}{\tau_2} \left( \hat{\Psi}_2 - \frac{L_h}{L_1} \hat{\Psi}_1 \right), \end{aligned} \quad (1.3.8)$$

kde  $\tau_1 = \sigma L_1 / R_1$  je časová konstanta statoru při rotorovém vinutí nakrátko a  $\tau_2 = \sigma L_2 / R_2$  je časová konstanta rotoru při statorovém vinutí nakrátko.

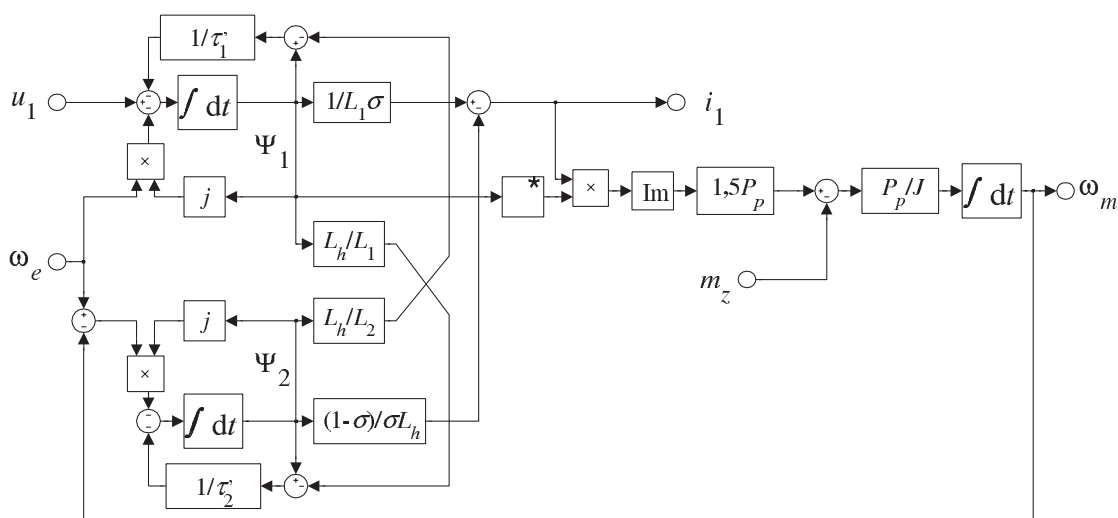
Nyní chybí pouze rovnice pro okamžitý moment motoru. Při odvození se vychází z energetické bilance motoru, kdy celková vnitřní energie motoru je rovna součtu mechanické a magnetické energie. Pokud bude modul spřaženého magnetického toku konstantní, je možno napsat na souřadnicovém systému nezávislý vztah

$$m = \frac{3}{2} P_p \operatorname{Im}[\hat{\Psi}_1^* \hat{i}_1] = -\frac{3}{2} P_p \operatorname{Im}[\hat{\Psi}_2^* \hat{i}_2], \quad (1.3.9)$$

kde  $P_p$  je počet pólových párů. Spolu s pohybovou rovnicí rotačních hmot

$$m - m_z = \frac{J}{P_p} \frac{d\omega_m}{dt} \Rightarrow \omega_m = \frac{P_p}{J} \int_0^t (m - m_z) dt + \omega_{m0}, \quad (1.3.10)$$

kde  $m_z$  je zátěžný moment a  $J$  je moment setrvačnosti, jsou odvozeny všechny rovnice nutné pro vyjádření matematického modelu asynchronního motoru. Stavový popis systému je tedy sestaven z rovnic (1.3.7), (1.3.8), (1.3.9) a (1.3.10). Blokové schéma matematického modelu je na obrázku Obr. 3.



**Obr. 3: Matematický model asynchronního motoru [1]**

## 2 VEKTOROVÉ ŘÍZENÍ

Způsobů řízení otáček asynchronního motoru existuje celá řada. Krom již zastaralých metod, jako změna počtu pólových dvojic nebo energeticky nevýhodná změna skluzu, je dnes často užíváno frekvenčního řízení. Jeho podstatou je držení stálého poměru  $U_1/\omega_e$ , což zajistí konstantní statorový magnetický tok a maximální moment. Nevýhodou je omezení platnosti pouze na ustálený stav. V úlohách náročných na dynamiku podává vektorové řízení lepší výsledky.

Jak již bylo zmiňováno, řízení otáček asynchronních motorů je, ve srovnání s jinými elektrickými motory, poměrně složité. Je to především díky faktu, že magnetický tok je funkcí zátěžného momentu a mechanických otáček. Přitom je k dispozici pouze jeden řídicí vstup. Vhodnými matematickými úpravami je však možno oddělit řízení magnetického toku a momentu motoru, čímž asynchronní motor získává vlastnosti stejnosměrného motoru s cizím buzením.

Vektorové řízení může být orientováno na hlavní, statorový či rotorový magnetický tok. Nejvýhodnější je orientace na rotorový tok, neboť umožňuje plné oddělení řízení momentu a magnetického toku [1]. Nutností je však znalost vektoru magnetického toku a úhlové rychlosti rotoru. Tok je možno měřit například pomocí Hallových sond (či snímačů magnetického pole obecně) umístěných v motoru. To však přináší nevýhody, jako snížení spolehlivosti a zvýšení pořizovacích nákladů. Lepší variantou je určení toku nepřímou, pomocí výpočtu. Většinou je počítán ze znalosti statorového napětí, proudu a otáček rotoru. Vybrané způsoby jeho určení budou rovněž uvedeny.

### 2.1 Odvození vektorového řízení

Odvození bude provedeno pro vektorové řízení orientované na rotorový magnetický tok. Odtud dále, pokud nebude řečeno jinak, bude magnetickým tokem míněn rotorový magnetický tok. Jako první je uvažován rotující  $d$ - $q$  souřadnicový systém, jehož reálná osa je spojena s vektorem rotorového magnetického toku  $\hat{\Psi}_2$ , který rotuje rychlostí  $\omega_f$ . To znamená, že  $q$ -složka toku bude vždy nulová. Pak při konstantním magnetickém toku můžeme z rovnice (1.3.4) vyjádřit vztah

$$0 = R_2 \hat{i}_2 + j(\omega_f - \omega_m) \hat{\Psi}_2 = R_2 \hat{i}_2 + js \omega_f \hat{\Psi}_2. \quad (2.1.1)$$

Po rozkladu do  $d$ - $q$  složek pak dostáváme

$$\begin{aligned} 0 &= R_2 (i_{2d} + ji_{2q}) + js \omega_f (\Psi_{2d} + j\Psi_{2q}) \\ 0 &= (R_2 i_{2d} - s \omega_f \Psi_{2q}) + j(R_2 i_{2q} + s \omega_f \Psi_{2d}). \end{aligned} \quad (2.1.2)$$

To samé je provedeno pro rovnici rotorového magnetického toku ve vztahu (1.3.5), tedy

$$\begin{aligned}\hat{\Psi}_2 &= L_2 \hat{i}_2 + L_h \hat{i}_1 = L_2 (i_{2d} + j i_{2q}) + L_h (i_{1d} + j i_{1q}) \\ \Psi_{2d} &= L_2 i_{2d} + L_h i_{1d} \\ \Psi_{2q} &= L_2 i_{2q} + L_h i_{1q} = 0.\end{aligned}\tag{2.1.3}$$

Nulová složka magnetického toku  $\Psi_{2q}$  zajistí, podle rovnice (2.1.2), nulový proud  $i_{2d}$ . Z rovnice (2.1.3) pak vyplývá

$$\Psi_{2d} = L_h i_{1d}.\tag{2.1.4}$$

Proudem  $i_{1d}$  je tedy možno přímo ovlivňovat velikost magnetického toku. Z rovnice (2.1.2) a (2.1.4) lze vyjádřit

$$i_{1q} = s \omega_f \frac{L_2}{R_2} i_{1d}.\tag{2.1.5}$$

Tento vztah bude později důležitý pro výpočet úhlové rychlosti magnetického pole  $\omega_f$ .

Nyní bude z rovnic (1.3.9) a (2.1.4) vyjádřen moment, tedy

$$\begin{aligned}m &= -\frac{3}{2} P_p \operatorname{Im}[\hat{\Psi}_2^* \hat{i}_2] = -\frac{3}{2} P_p \operatorname{Im}[(\Psi_{2d} - j \Psi_{2q})(i_{2d} + j i_{2q})] = \\ &= \frac{3}{2} P_p (\Psi_{2q} i_{2d} - \Psi_{2d} i_{2q}) = \frac{3}{2} P_p \frac{L_h}{L_2} \Psi_{2d} i_{1q}.\end{aligned}\tag{2.1.6}$$

Ze vztahů (2.1.4) a (2.1.6) lze pozorovat, že bylo dosaženo zmiňovaného oddělení řízení magnetického toku a momentu. Odtud je proud  $i_{1d}$  nazýván tokotvorný a proud  $i_{1q}$  momentotvorný. Zároveň lze vidět, že pokud je dodržena podmínka konstantního magnetického toku, jedná se o lineární vztahy.

## 2.2 Odvození přenosů nutných pro návrh regulátorů

Pro návrh regulátorů zbývá odvodit některé přenosy asynchronního motoru.

## 2.2.1 Proudový přenos

Vstupem soustavy je statorové napětí a výstupem statorový proud. Nejprve je třeba rozložit rovnici statorového napětí (1.3.4) na imaginární a reálnou část v  $d$ - $q$  složkách, tedy

$$\begin{aligned} u_{1d} + ju_{1q} &= R_1(i_{1d} + ji_{1q}) + \frac{d}{dt}(\hat{\Psi}_{1d} + j\hat{\Psi}_{1q}) + j\omega_f(\hat{\Psi}_{1d} + j\hat{\Psi}_{1q}) \\ u_{1d} &= R_1i_{1d} + \frac{d}{dt}\hat{\Psi}_{1d} - \omega_f\hat{\Psi}_{1q} \\ u_{1q} &= R_1i_{1q} + \frac{d}{dt}\hat{\Psi}_{1q} + \omega_f\hat{\Psi}_{1d}. \end{aligned} \quad (2.2.1)$$

To samé je provedeno pro rovnici statorového toku (1.3.4)

$$\begin{aligned} \Psi_{1d} + j\Psi_{1q} &= L_1(i_{1d} + ji_{1q}) + L_h(i_{2d} + ji_{2q}) \\ \Psi_{1d} &= L_1i_{1d} + L_hi_{2d} = L_1i_{1d} \\ \Psi_{1q} &= L_1i_{1q} + L_hi_{2q} = \left(L_1 - \frac{L_h^2}{L_2}\right)i_{1q} = \sigma L_1i_{1q}. \end{aligned} \quad (2.2.2)$$

Po dosazení rovnic (2.2.2) do rovnic (2.2.1) lze získat vztahy

$$\begin{aligned} u_{1d} &= R_1i_{1d} + L_1 \frac{di_{1d}}{dt} - \sigma L_1 \omega_f i_{1q} = u_{1d}^{lin} + u_{1d}^{nlin} \\ u_{1q} &= R_1i_{1q} + \sigma L_1 \frac{di_{1q}}{dt} + L_1 \omega_f i_{1d} = u_{1q}^{lin} + u_{1q}^{nlin}, \end{aligned} \quad (2.2.3)$$

které lze rozdělit na lineární a nelineární část, tedy

$$u_{1d}^{lin} = R_1i_{1d} + L_1 \frac{di_{1d}}{dt} \quad (2.2.4)$$

$$\begin{aligned} u_{1q}^{lin} &= R_1i_{1q} + \sigma L_1 \frac{di_{1q}}{dt} \\ u_{1d}^{nlin} &= -\sigma L_1 \omega_f i_{1q} \\ u_{1q}^{nlin} &= L_1 \omega_f i_{1d}. \end{aligned} \quad (2.2.5)$$



Aby byly získány lineární proudové přenosy, bude nelineární část ignorována. Tu pak mohou odstranit integrační složky regulátorů proudu. Avšak lepším řešením je nelineární složky počítat a následně k akčním zásahům regulátorů přičíst. To se obvykle označuje jako *decoupling*. Nyní je možno z rovnic (2.2.4) psát operátorové přenosy

$$\begin{aligned} F_{id}(p) &= \frac{I_{1d}(p)}{U_{1d}^{lin}(p)} = \frac{1}{L_1 p + R_1} = \frac{\frac{1}{R_1}}{\tau_{1d} p + 1} & \tau_{1d} &= \frac{L_1}{R_1} \\ F_{iq}(p) &= \frac{I_{1q}(p)}{U_{1q}^{lin}(p)} = \frac{1}{\sigma L_1 p + R_1} = \frac{\frac{1}{R_1}}{\tau_{1q} p + 1} & \tau_{1q} &= \frac{\sigma L_1}{R_1}, \end{aligned} \quad (2.2.6)$$

kde  $\tau_{1d}$  je časová konstanta přenosu proudu  $i_{1d}$  a  $\tau_{1q}$  je časová konstanta přenosu proudu  $i_{1q}$ .

## 2.2.2 Přenos otáček a magnetického toku $\Psi_{2d}$

Přenos  $\Psi_{2d}$  je získán z rovnice (2.1.4). Platí

$$F_{\Psi_{2d}}(p) = \frac{\Psi_{2d}(p)}{I_{1d}(p)} = L_h. \quad (2.2.7)$$

Vstupní veličinou otáčkového přenosu je proud  $i_{1q}$ , výstupem pak mechanické otáčky  $\omega_m$ . Při zanedbání zátěžného momentu  $m_z$ , který je uvažován jako porucha, dostáváme z rovnice (2.1.6) vztah

$$\frac{d\omega_m}{dt} = \frac{m}{J} = \frac{3L_h \Psi_{2d}}{2JL_2} i_{1q}. \quad (2.2.8)$$

Vzhledem k tomu, že magnetický tok je držen konstantní, jedná se o lineární diferenciální rovnici. Výsledný přenos je

$$F_{\omega_m}(p) = \frac{\Omega_m(p)}{I_{1q}(p)} = \frac{3L_h \Psi_{2d}}{2JL_2} \frac{1}{p}. \quad (2.2.9)$$

## 2.3 Způsob návrhu regulátorů

Pro regulaci proudu jsou obvykle voleny diskrétní PI regulátory. V našem případě regulátory v paralelním tvaru s přenosem

$$F_{PI}(z) = K_{PI} \left( 1 + \frac{T_s}{T_i} \frac{z^{-1}}{1 - z^{-1}} \right), \quad (2.3.1)$$

kde  $K_{PI}$  je zesílení,  $T_s$  vzorkovací perioda a  $T_i$  integrační časová konstanta. Obvykle se nastavují tak, aby se nula v přenosu regulátoru vykrátila s pólem v přenosu soustavy. Proudové přenosy řízení jsou potom prvního řádu a pokud je vzorkovací perioda dostatečně krátká, nemůže dojít k rozkmitání. Je však třeba mít na paměti, že všechny odvozené přenosy jsou platné pouze při konstantním magnetickém toku, což při odbuzování nemusí po krátkou dobu platit. Fázová, ani amplitudová bezpečnost tedy nemusí být garantována.

Regulátor proudu  $i_{1d}$  je obvykle limitován na horní mez statorového napětí  $U_1^{max}$ . Regulátor  $i_{1q}$  bývá omezen na hodnotu danou vztahem

$$u_{1q}^{max} = \sqrt{(U_1^{max})^2 - u_{1d}^2}. \quad (2.3.2)$$

Regulátor magnetického toku poskytuje referenci regulátoru proudu  $i_{1d}$ . Obvykle se volí jako diskrétní PID regulátor. V našem případě v paralelním tvaru, s filtrací derivační složky, dle přenosu

$$F_{PID}(z) = K_{PID} \left( 1 + \frac{T_s}{T_i} \frac{z^{-1}}{1 - z^{-1}} + N \frac{1 - z^{-1}}{1 - e^{-\frac{T_s N}{T_d}} z^{-1}} \right), \quad (2.3.3)$$

kde  $K_{PID}$  je zesílení,  $T_s$  je vzorkovací perioda,  $T_i$  integrační časová konstanta,  $T_d$  derivační časová konstanta a  $N$  filtrační konstanta obvykle volená v rozsahu 3 až 20 [5]. Akční zásah regulátoru  $\Psi_{2d}$  bývá limitován horní mezí statorového proudu  $I_1^{max}$ .

Regulátor otáček je volen jako diskrétní PID regulátor, v tomto případě opět dle přenosu 2.3.3. Limitace akčního zásahu je provedena dle vztahu

$$i_{1q}^{max} = \sqrt{(I_1^{max})^2 - (i_{1d}^{ref})^2}, \quad (2.3.4)$$

kde  $i_{1q}^{max}$  je maximální hodnota akčního zásahu a  $i_{1d}^{ref}$  je žádaná hodnota regulátoru proudu  $i_{1d}$ .

Pro zkrácení doby výpočtu bývá obvyklé, že je akční zásah počítán s vzorkovací periodou až osmkrát delší než zbylé části algoritmu vektorového řízení.

Nutností je implementace ochrany integrátoru proti přetečení (anti-windup) u všech regulátorů. V našem případě je od integrátoru odečítán rozdíl mezi limitovaným a nelimitovaným akčním zásahem, násobeným volenou konstantou  $T_i$ . Výhodou tohoto způsobu je, že neobsahuje algebraickou smyčku.

## 2.4 Výpočet magnetického toku $\Psi_{2d}$ a jeho úhlové rychlosti

Výpočet rotorového magnetického toku je možno provést mnoha způsoby. Zmíněny budou pouze některé jednoduché metody [6].

Nejjednodušší je výpočet z napěťového modelu asynchronního motoru ve statorových souřadnicích. Pro rotorový tok pak platí

$$\hat{\Psi}_2 = \frac{L_2}{L_h} \int_0^t (\hat{u}_1 - R_1 \hat{i}_1) dt + \frac{L_h^2 - L_1 L_2}{L_h} \hat{i}_1. \quad (2.4.1)$$

Jedná se tedy o jednoduchou integraci, což může přinést potíže s driftem integrátoru. Důvodem je neznalost přesné hodnoty odporu, což je znatelné především při nízkém statorovém napětí či vysokém statorovém proudu. Ve vysokých otáčkách je statorové napětí dostatečně velké, takže se drift již tolik neprojevuje.

Další možností je výpočet z proudového modelu, čemuž odpovídá vztah

$$\frac{d\hat{\Psi}_2}{dt} = \frac{R_2 L_h}{L_2} \hat{i}_1 - \hat{\Psi}_2 \left( \frac{R_2}{L_2} - j P_p \omega_m \right). \quad (2.4.2)$$

Hlavními nevýhodami této metody je citlivost na chybu hodnoty rotorového odporu a nutnost znalosti rotorových otáček. Problém s driftem integrátoru je však odstraněn.

V praxi užívanou metodou je výpočet z kombinovaného modelu. Platí vztah

$$\frac{d\hat{\Psi}_2}{dt} = \frac{R_2 L_h L_2}{L_h^2 R_2 + L_2^2 R_1} \hat{u}_1 - \frac{R_1 L_2^2}{L_h^2 R_2 + L_2^2 R_1} \left( \frac{R_2}{L_2} - j P_p \omega_m \right) \hat{\Psi}_2 - \frac{R_2 L_h (L_1 L_2 - L_h^2)}{L_h^2 R_2 + L_2^2 R_1} \frac{d\hat{i}_1}{dt}. \quad (2.4.3)$$

Ani zde nedochází k driftu integrátoru. Výhodou je malá citlivost vůči změně parametrů. Opět je nutná znalost mechanických otáček a rotorového odporu. Problém může působit čistá derivace statorového proudu. V praktické aplikaci je tedy občas nutno použít filtr.

Rovněž otáčky rotorového magnetického toku  $\omega_f$  lze počítat různými způsoby. Jednoduchý a v praxi používaný vztah je odvozen z rovnice (2.1.5), tedy

$$\omega_f = \frac{R_2 i_{1q}}{L_2 i_{1d}} + P_p \omega_m. \quad (2.4.4)$$

Problémem však je, že při velmi nízkém proudu  $i_{1d}$  (tedy po zapnutí motoru, kdy je rotorový magnetický tok nulový) se ve výpočtu nachází singularita. Tento problém lze jednoduše odstranit tak, že do určité malé velikosti proudu  $i_{1d}$  uvažujeme namísto úhlové rychlosti  $\omega_f$ , rychlost  $P_p \omega_m$ . Chyba je velice malá, neboť zpočátku si jsou obě rychlosti velice blízké. Následně, po ustálení žádaného magnetického toku  $\Psi_{2d}$ , již k problému nedochází.

## 2.5 Výsledná struktura vektorového řízení

Matematický model vektorového řízení, navržený v prostředí MATLAB-Simulink verze R2008b je v příloze 1. Jako matematický model asynchronního motoru byl použit AC Induction Motor Block z knihovny Electromechanical Drives Toolbox [7]. Magnetický tok  $\Psi_{2d}$  je počítán z kombinovaného modelu podle rovnice (2.4.3) a jeho úhlová rychlost podle rovnice (2.4.4).

Ve schématu chybí model PWM napětí, neboť nijak neovlivňuje výsledky simulace a pouze by prodlužoval výpočet.

Požadavek na roztočení motoru po jeho zapnutí musí přijít až po určité minimální době, během které je vytvořen a ustálen požadovaný magnetický tok. Do té doby neplatí vztahy odvozené pro vektorové řízení.

### 3 VYBRANÉ ALGORITMY ODBUZOVÁNÍ

Pokud zvyšujeme otáčky asynchronního motoru, roste i zpětné indukované napětí, které musí být dodáváno zdrojem. V určitém okamžiku je tak dosaženo maximálního napětí zdroje a otáčky není možno dále zvyšovat. Tedy pokud je magnetický tok ponechán stávající hodnotě. Výslednou amplitudu statorového napětí získáme ze vztahu

$$u_1 = \sqrt{u_{1d}^2 + u_{1q}^2} . \quad (3.1)$$

V nelineární části napěťové rovnice (2.2.5) pro  $u_{1q}$  se nachází součin úhlové rychlosti pole  $\omega_f$  a proudu  $i_{1d}$  (ten je přímo úměrný magnetickému toku  $\Psi_{2d}$ ). Je jasné, že pro vysoké otáčky tento součin nabývá značných hodnot. Avšak odbuzením, čili zmenšením proudu  $i_{1d}$ , je nelineární přírůstek napětí zmenšen a je možno dosáhnout vyšších otáček. Při provozu asynchronního stroje na otáčkách nižších než jmenovitých je vhodné držet magnetický tok na maximální hodnotě, která je stejná jako při napájení síťovým napětím. Tím je zajištěno maximální magnetické využití stroje.

Nelineární  $d$ -složka statorového napětí, ve srovnání s  $q$ -složkou potíže nečiní, neboť je násobena činitelem rozptylu  $\sigma$ , který bývá mnohem menší než jedna.

Nevýhodou odbuzování je ztráta maximálního momentu, neboť součin magnetického toku  $\Psi_{2d}$  a maximální  $q$ -složky proudu v rovnici (2.1.6) bude menší.

Algoritmy odbuzování pak, obvykle s ohledem na velikost statorového napětí či frekvence pole  $\omega_f$ , nastavují žádanou hodnotu magnetického toku. V této kapitole budou uvedeny čtyři algoritmy.

#### 3.1 Algoritmus 1/ $\omega_f$

Jedná se o nejjednodušší ze srovnávaných algoritmů. Princip je založen na jednoduchém vztahu

$$\Psi_{2d}^{ref} = \frac{\omega_{base}}{|\omega_f|} \Psi_{2d}^{max} , \quad (3.1.1)$$

kde  $\Psi_{2d}^{max}$  je zvolená maximální hodnota magnetického toku,  $\Psi_{2d}^{ref}$  je žádaná hodnota magnetického toku a  $\omega_{base}$  je úhlová rychlost magnetického pole, při které se přechází do regionu odbuzování [8]. Maximální a minimální hodnota magnetického toku je samozřejmě omezena. Při výpočtu  $\omega_{base}$  se vychází z nelineárních rovnic (2.2.5). Je tedy uvažován pouze ustálený stav a zároveň zanedbány úbytky na statorovém a rotorovém odporu. To si lze dovolit, neboť při vysokých otáčkách jsou lineární části napěťových

rovnice (2.2.4), ve srovnání s nelineární částí, zanedbatelné. Po dosazení do rovnice (3.1) pak platí

$$\begin{aligned} (U_1^{max})^2 &\geq (u_{1d}^{nelin})^2 + (u_{1q}^{nelin})^2 \\ (U_1^{max})^2 &\geq (\sigma L_1 \omega_f i_{1q})^2 + (L_1 \omega_f i_{1d})^2. \end{aligned} \quad (3.1.2)$$

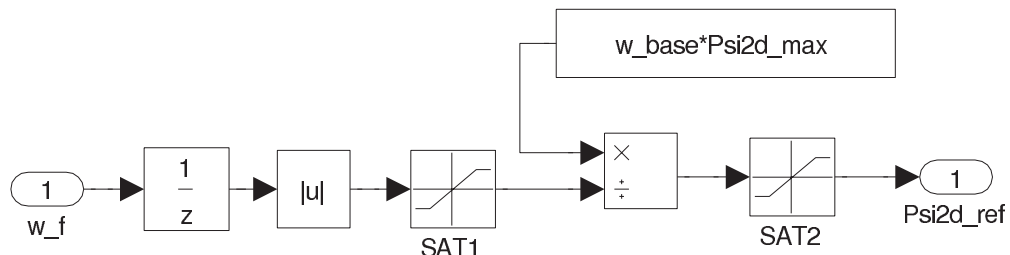
Nyní bude z této nerovnice odvozena  $\omega_{base}$ . Ta je získána jako úhlová frekvence, při které je dosaženo maximálního statorového napětí, tedy

$$(\sigma L_1 \omega_{base} i_{1q})^2 + (L_1 \omega_{base} i_{1d})^2 = (U_1^{max})^2. \quad (3.1.3)$$

Avšak vlivem všech zanedbání je takováto hodnota  $\omega_{base}$  příliš vysoká a k odbuzování dochází příliš pomalu či vůbec. Proto je zaveden volitelný koeficient  $k_{1/\omega_f}$ , jehož hodnota je nastavována nižší, blízká jedné. Budeme-li pokračovat v odvozování, dostaneme

$$\begin{aligned} L_1^2 \omega_{base}^2 \left\{ \sigma^2 \left[ (I_1^{max})^2 - \left( \frac{\Psi_{2d}^{max}}{L_h} \right)^2 \right] + \left( \frac{\Psi_{2d}^{max}}{L_h} \right)^2 \right\} &= (k_{1/\omega_f} U_1^{max})^2 \\ \omega_{base} &= \frac{k_{1/\omega_f} L_h U_1^{max}}{L_1 \sqrt{(\Psi_{2d}^{max})^2 (1 - \sigma^2) + (\sigma L_h I_1^{max})^2}}. \end{aligned} \quad (3.1.4)$$

Schéma realizace v prostředí MATLAB-Simulink je na obrázku Obr. 4. Blok saturace SAT1 omezuje dolní mez  $\omega_f$ , čímž se předchází dělení nulou. Saturace SAT2 pak omezuje referenci magnetického toku do námi požadovaného intervalu. Zpoždění o jeden vzorek zde odstraňuje algebraickou smyčku, která by mohla vzniknout, jelikož algoritmus odbuzování je obvykle počítán dříve než úhlová rychlost  $\omega_f$ .

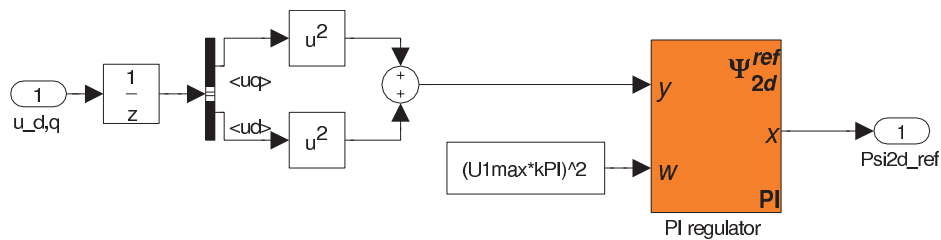


**Obr. 4: Schéma algoritmu odbuzování  $1/\omega_f$  v prostředí MATLAB-Simulink**

### 3.2 Odbuzování s PI regulátorem

Jak už název napovídá, hlavní částí algoritmu je PI regulátor. Jeho úkolem je regulovat amplitudu statorového napětí prostřednictvím reference magnetického toku  $\Psi_{2d}$  jako akčního zásahu. Funkce je nejlépe patrná ze schématu na obrázku Obr 5. Z důvodu zjednodušení algoritmu, tedy odstranění odmocniny ve vztahu 3.1, se počítá s druhou mocninou amplitudy statorového napětí jako regulované veličiny. Výstup regulátoru je samozřejmě omezen na požadovaný rozsah magnetického toku. To vyžaduje přítomnost anti-windup ochrany.

Opět jako u algoritmu  $1/\omega_f$  je nutné začít odbuzovat o něco dříve. Respektive žádaná hodnota statorového napětí musí být nižší než jeho maximální hodnota. Nikdy by totiž nebyla získána záporná regulační odchylka. Opět tedy zavádíme koeficient  $k_{PI}$ . Jeho hodnotu je třeba volit dostatečně blízko jedné. V opačném případě by byl magnetický tok zbytečně malý, což by snížilo maximální moment. Algebraická smyčka je odstraněna zpožděním výpočtu o jeden vzorek.



Obr. 5: Schéma algoritmu odbuzování s PI regulátorem v prostředí MATLAB-Simulink

### 3.3 Odbuzování na maximální moment

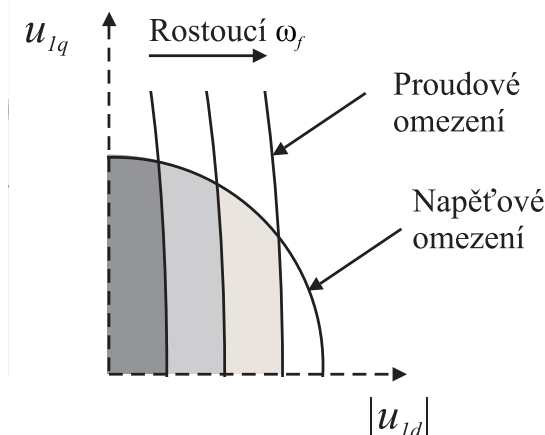
Nejprve uveďme rovnice napěťového a proudového omezení [8], [9]

$$\begin{aligned} u_{1d}^2 + u_{1q}^2 &\leq (U_1^{max})^2 \\ i_{1d}^2 + i_{1q}^2 &\leq (I_1^{max})^2. \end{aligned} \quad (3.3.1)$$

Stejně jako u algoritmu  $1/\omega_f$  jsou zavedeny zjednodušující předpoklady pro napěťové rovnice (2.2.3). Je uvažován pouze ustálený stav a napěťové úbytky na rotorovém a statorovém odporu jsou zanedbány. V úvahu je brána pouze nelineární část, tedy rovnice (2.2.5). Potom proudové omezení pomocí napěťových složek je

$$\left( \frac{u_{1d}}{\sigma L_1 \omega_f} \right)^2 + \left( \frac{u_{1q}}{L_1 \omega_f} \right)^2 \leq (I_1^{max})^2. \quad (3.3.2)$$

Grafická interpretace rovnice (3.3.2), v  $d$ - $q$  složkách statorového napětí, je obsah elipsy, jež se zvětšuje s rostoucí uhlovou rychlostí  $\omega_f$ . Napět'ové omezení, dle rovnice (3.3.1), lze zobrazit jako obsah kruhu. Společná plocha obou kuželoseček udává celkové napět'ové omezení, v němž se musí nacházet výsledný vektor statorového napětí. Situace je znázorněna na obrázku Obr. 6. Při náběhu motoru je  $\omega_f$  velice malá a elipsa proudového omezení je menší než kruh napět'ového omezení. Kružnice napět'ového omezení je dosaženo až s vyššími otáčkami. Tehdy nastává počátek oblasti odbuzování.



Obr. 6: Napět'ové omezení v napět'ových souřadnicích [9]

Nyní bude momentová rovnice (2.1.6) vyjádřena v napět'ových souřadnicích, při uvážení všech zmíněných zjednodušení. Lze psát [9]

$$m = \frac{3P_p L_h^2}{2L_2} i_{1d} i_{1q} = -\frac{3P_p L_h^2}{2L_2} \frac{u_{1d}}{\sigma L_1 \omega_f} \frac{u_{1q}}{L_1 \omega_f} = K u_{1d} u_{1q}, \quad (3.3.3)$$

kde  $K$  je

$$K = -\frac{3P_p L_h^2}{2L_2 \sigma L_1^2 \omega_f^2}.$$

Pro danou frekvenci  $\omega_f$  se jedná o konstantu. Můžeme tedy zavést novou funkci

$$F = \frac{m}{K} = u_{1d} u_{1q}, \quad (3.3.4)$$

kteřá pro konstantní  $\omega_f$  reprezentuje hyperbolu. Při geometrické interpretaci se její vrchol s rostoucí  $\omega_f$  vzdaluje od počátku souřadnic. Lze tedy předpokládat, že pokud je, s ohledem na proudové a napět'ové omezení, dosaženo maximální hodnoty dané funkcí  $F$ , je docíleno i maximálního momentu.

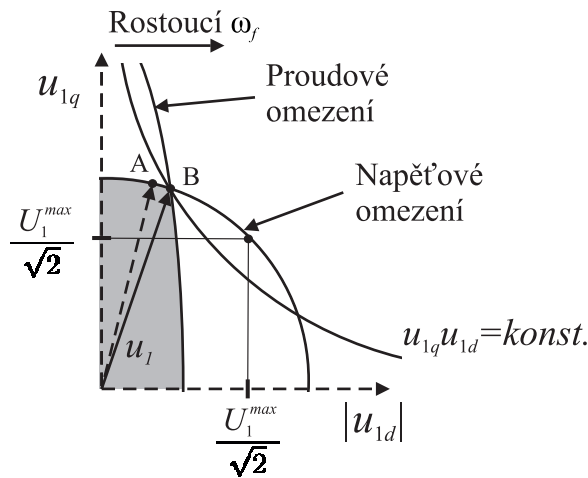


V případě, že se vrchol hyperboly nachází mimo kružnici napěťového omezení, je maximálního momentu dosaženo v bodě

$$|u_{1d}| = u_{1d} = \frac{U_1^{max}}{\sqrt{2}}. \quad (3.3.5)$$

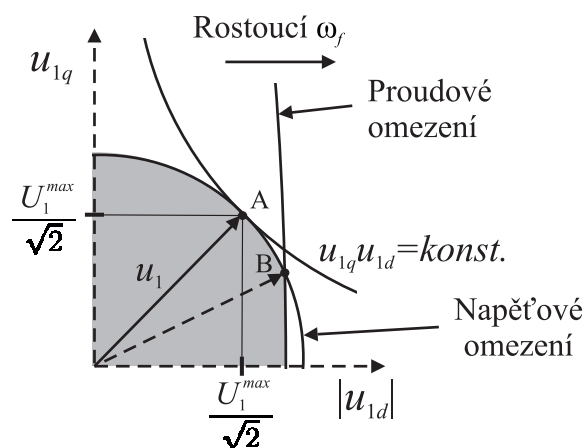
Celou oblast  $\omega_f$ , ve které je motor odbuzován, je možno rozdělit na dva regiony. Úhlovou rychlost  $\omega_f$ , při které se přechází z oblasti běžného provozu při maximálním toku do prvního regionu odbuzování, nazveme  $\omega_{base1}$  a rychlost  $\omega_f$ , při které se přechází z prvního regionu do druhého, nazveme  $\omega_{base2}$ .

Ve chvíli, kdy je dosaženo  $\omega_{base1}$ , se vektor statorového napětí téměř dotýká kružnice napěťového omezení. Situace je znázorněna na obrázku Obr. 7, kde je krom napěťového a proudového omezení vyznačena také funkce  $F$  a optimální vektor statorového napětí. Bez odbuzování je pracovním bodem bod A. Aby bylo dosaženo maximálního momentu, je nutné snížit napětí  $u_{1q}$  (odpovídá snížení proudu  $i_{1d}$  a tedy i magnetického toku) a přesunout se do optimálního pracovního bodu B. V regionu 1 se tedy vektor  $U_1$  pohybuje na průsečku napěťového a proudového omezení.



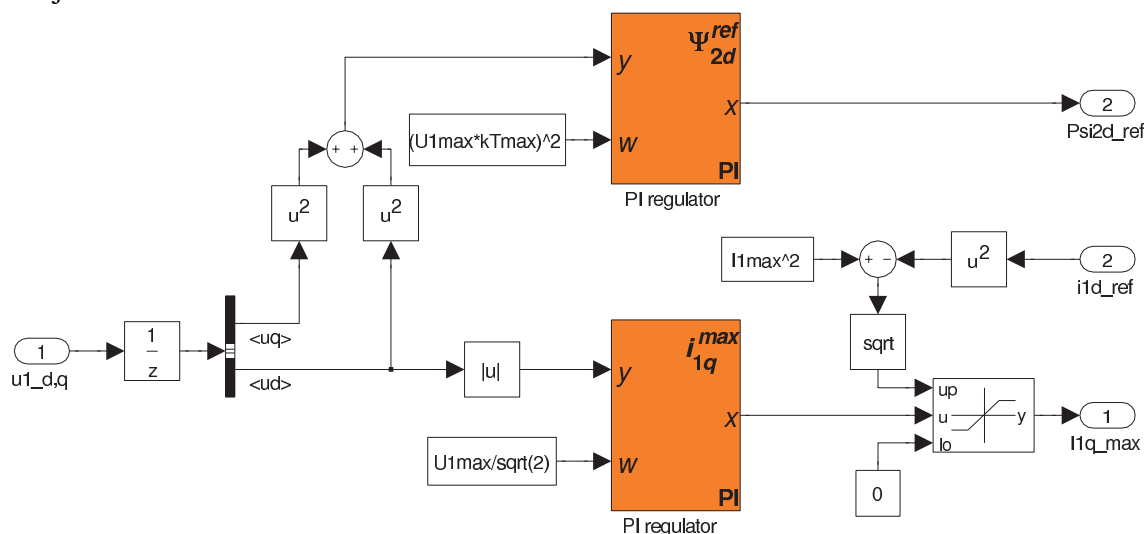
Obr. 7: Optimální napěťový vektor, při odbuzování v prvním regionu [9]

Pokud bude  $\omega_f$  dále zvyšována, přesáhne nakonec elipsa proudového omezení i bod popsany rovnicí (3.3.5) (bod A na obrázku Obr. 8) a je dosaženo druhého regionu odbuzování. Pokud by pracovní bod i nadále sledoval průsečík napěťového a proudového omezení v bodě B, bude funkce  $F$  dosahovat menších hodnot než v optimálním pracovním bodě A. Je tedy třeba omezovat napětí  $u_{1d}$  (respektive proud  $i_{1q}$ ) a zároveň zvyšovat napětí  $u_{1q}$ , tedy proud  $i_{1d}$ . Magnetický tok je tak držen na hodnotě vyšší než v bodě B a tím je dosaženo vyššího momentu.



Obr. 8: Optimální napět'ový vektor, při odbuzování v druhém regionu [9]

Schéma algoritmu v prostředí MATLAB-Simulink je na obrázku Obr. 9. Odbuzování v regionu 1 je realizováno pomocí PI regulátoru. Je možno si povšimnout, že tato část je shodná s algoritmem odbuzování pomocí PI regulátoru popsaného v předchozí kapitole. A to včetně zavedení koeficientu  $k_{Tmax}$ , pro něhož platí vše uvedené v kapitole 3.2. V regionu 2 začne být omezován i proud  $i_{1q}$ , o což se opět stará PI regulátor. Vztah (2.3.4) samozřejmě platí rovněž. Použita je pak menší z hodnot  $i_{1q}^{max}$ . To je naznačeno i na obrázku Obr. 9.



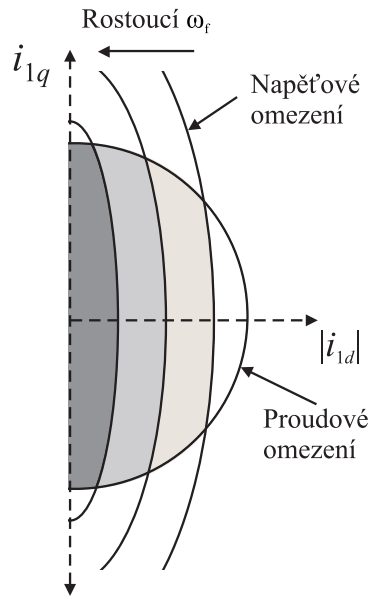
Obr. 9: Realizace algoritmu odbuzování na maximální moment v MATLAB-Simulink

### 3.4 Algebraické odbuzování na maximální moment

Teoretický základ je stejný jako v předchozí kapitole 3.3. Tentokrát však bude použito proudových souřadnic [8]. Výsledný algoritmus tak nebude obsahovat regulátory a vše bude počítáno podle příslušných vztahů. Vyjde-li se z rovnic (3.3.1), jsou získány nerovnice pro omezení proudu a napětí

$$\begin{aligned} (I_1^{max})^2 &\geq i_{1d}^2 + i_{1q}^2 \\ (U_1^{max})^2 &\geq (\sigma L_1 \omega_f i_{1q})^2 + (L_1 \omega_f i_{1d})^2. \end{aligned} \quad (3.4.1)$$

Grafická interpretace proudového omezení v proudových souřadnicích je kruh a napěťového omezení elipsa, která se zmenšuje s rostoucí  $\omega_f$ . Znázornění je na obrázku Obr. 10.



**Obr. 10: Napěťové a proudové omezení v proudových souřadnicích**

Oblast  $\omega_f$ , kde dochází k odbuzování, je opět rozdělena na dva regiony stejně jako v kapitole 3.3. Úhlová rychlost na počátku prvního regionu je nazvána  $\omega_{base1}$  a na počátku druhého regionu  $\omega_{base2}$ .

Úhlovou rychlost  $\omega_{base1}$  lze vypočítat podle vztahu

$$\omega_{base1} = \frac{\sqrt{\left(U_1^{max} L_1 I_{1d}\right)^2 + \left(R_1 I_{1q}\right)^2} - R_1 I_{1q}}{L_1 I_{1d}}, \quad (3.4.2)$$

kde

$$I_{1d} = \frac{\Psi_{2d}^{max}}{L_h} \quad (3.4.3)$$

$$I_{1q} = \sqrt{\left(I_1^{max}\right)^2 - I_{1d}^2}.$$

Stejně jako v kapitole 3.3 je pracovní bod v prvním regionu držen v průsečíku proudového a napěťového omezení. Příslušnou  $d$ -složku statorového proudu (po vynásobení hlavní indukčnosti  $L_h$  udává referenci magnetického toku) lze určit dle vztahu

$$i_{1d} = \sqrt{\frac{\left(\frac{U_1^{max}}{\omega_f}\right)^2 - \left(\sigma L_1 I_1^{max}\right)^2}{L_1^2 - \sigma^2 L_1^2}}. \quad (3.4.4)$$

Maximální hodnota  $q$ -složky proudu je pak dána vztahem (2.3.4).

Do druhého regionu odbuzování je vstoupeno po překročení úhlové rychlosti

$$\omega_{base2} = \sqrt{\frac{1 + \sigma^2}{2\sigma^2 L_1^2}} \frac{U_1^{max}}{I_1^{max}}. \quad (3.4.5)$$

V druhém regionu je pracovní bod nastaven podle podmínky (3.3.5). V proudových souřadnicích pak platí

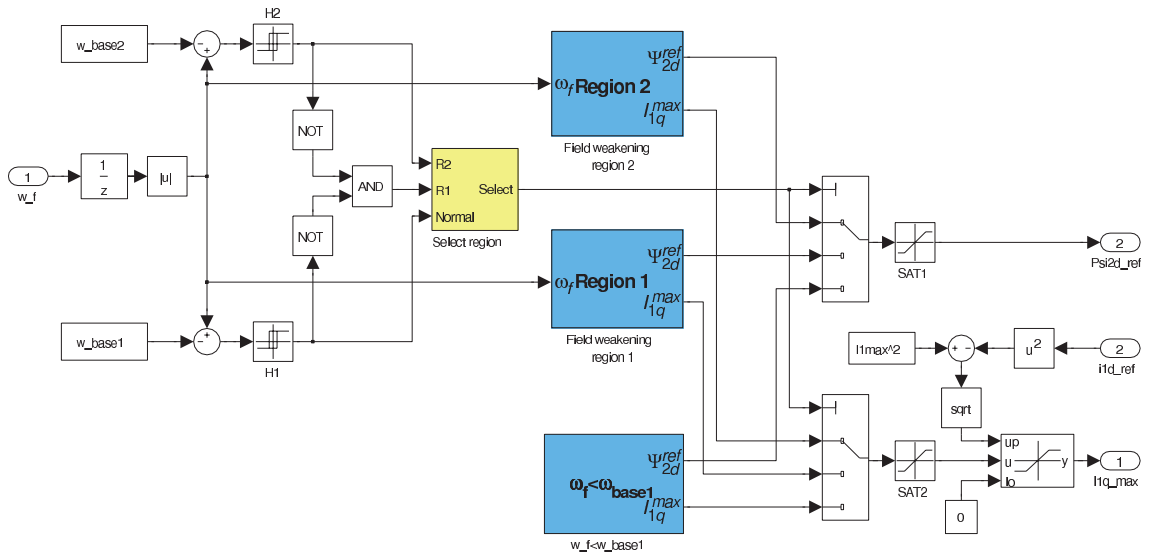
$$i_{1d}^{ref} = \frac{U_1^{max}}{\sqrt{2}\omega_f L_1} \quad (3.4.6)$$

$$i_{1q}^{max} = \frac{U_1^{max}}{\sqrt{2}\omega_f \sigma L_1},$$

přičemž vztah (2.3.4) zůstává platný. V potaz se bere nižší hodnota  $i_{1q}^{max}$ .

Stejně jako u algoritmu  $1/\omega_f$  je (vzhledem k zanedbáním, jež jsou uvedena v kapitole 3.3) nutno zavést koeficient  $k_{Algeb}$ , jímž je násobena hodnota  $U_1^{max}$  ve všech vztazích (3.4.2) až (3.4.8). Koeficient je opět volen nižší, blízký jedné.

Schéma algoritmu v programu MATLAB-Simulink je na obrázku Obr. 11. Podle hodnoty  $\omega_f$  jsou hodnoty  $\Psi_{2d}^{ref} = L_h i_{1d}^{ref}$  a  $i_{1q}^{max}$  nastavovány dle rovnic (3.4.6) (druhý region), (3.4.5) (první region) nebo na maximální hodnoty při  $\omega_f < \omega_{base1}$ . Pro odstranění případného kmitání na hranicích mezi regiony, byla do přepínací charakteristiky přidána hystereze (H1, H2). Tu není vhodné volit příliš velkou, neboť se snižuje účinnost odbuzování a navíc vztahy v blízkosti hranic regionů relativně navazují. Funkce vztahu (2.3.4) je rovněž znázorněna na obrázku Obr. 11, kdy je použita menší z výsledných hodnot  $i_{1q}^{max}$ .



Obr. 11: Realizace algoritmu odbuzování na maximální moment algebraicky v prostředí MATLAB-Simulink

## 4 IMPLEMENTACE ŘÍDICÍ ČÁSTI ALGORITMU DO DSC 56F8346

V této kapitole bude popsána implementace řídicí části algoritmu vektorového řízení a algoritmu odbuzování do mikroprocesoru s výpočty v pevné řádové čárce.

Jak již bylo zmíněno, k rozšíření vektorového řízení asynchronních motorů došlo až v nedávné době. Je to především díky tomu, že se jedná o výpočetně náročný algoritmus, který běžné mikroprocesory nezvládaly. Změnu přineslo až rozšíření digitálních signálových procesorů (DSP), umožněné poklesem jejich cen. DSP jsou speciálně uzpůsobeny pro zpracování signálů. Obvykle jsou založeny na Harvardské architektuře a obsahují velmi rychlé sčítačky, násobičky, bitové posuvy a mac (multiply and accumulate) instrukce. Pokud je DSP vybaven periferiemi, podobně jako mikrokontrolér, používá se pro něj název digitální signálový kontrolér (DSC)

Pro účely tohoto projektu byl zvolen DSC rodiny 56F800E firmy Freescale, konkrétně typ 56F8346. Obsahuje DSP druhé generace s výpočty v pevné řádové čárce, vhodný právě pro řízení motorů. Mezi nejdůležitější vlastnosti patří 16-bitová duální Harvardská architektura, umožňující paralelní čtení dvou operandů, výpočetní výkon až 60MIPS při 60MHz. Dále 8KB RAM pro data a 4KB RAM pro program, čtyři 36-bitové akumulátory a rychlé vícebitové posuvy [10], [11].

Velkou výhodou je rovněž softwarové vybavení, jež firma Freescale k tomuto DSC poskytuje. Krom vývojového prostředí CodeWarrior (zde verze 5.6.1.1658), jež obsahuje mnoho debugovacích nástrojů, je k dispozici řada knihoven. V tomto projektu se jedná především o `intrinsics_56800E.h`, která obsahuje intrinsické funkce. Ty umožňují efektivní využití výpočtů v pevné řádové čárce při použití jazyka C a tedy vyhnutí se assembleru. Další velmi užitečnou knihovnou je Motor Control Library (`mclib.h`) [12]. Ta obsahuje, krom definic mnoha datových typů, například goniometrické funkce, Clarkové a Parkovu transformaci, odmocninu a celou řadu dalších funkcí.

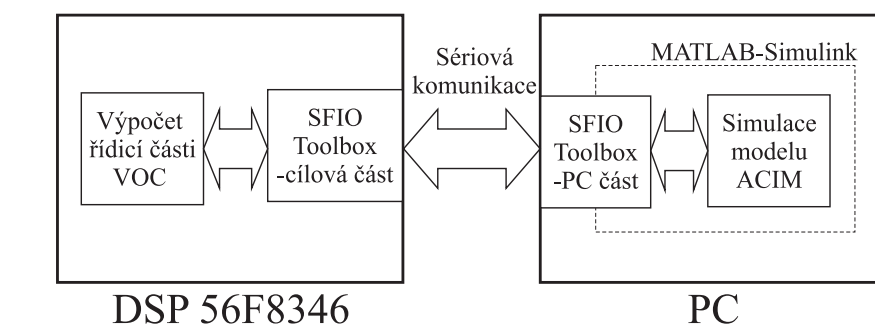
### 4.1 Metoda Processor-in-the-loop

Základem metody je přesunutí výpočtu požadované části simulace na jiný, cílový procesor. Je tedy možno algoritmy přesunovat postupně, po částech a výstupy následně srovnávat se simulací. To usnadňuje vývoj a odladění klíčových částí algoritmů, což by jinak představoval relativně náročný proces. Zvláště při výpočtech v pevné řádové čárce, kde hrozí nechtěné přetečení či saturace proměnné.

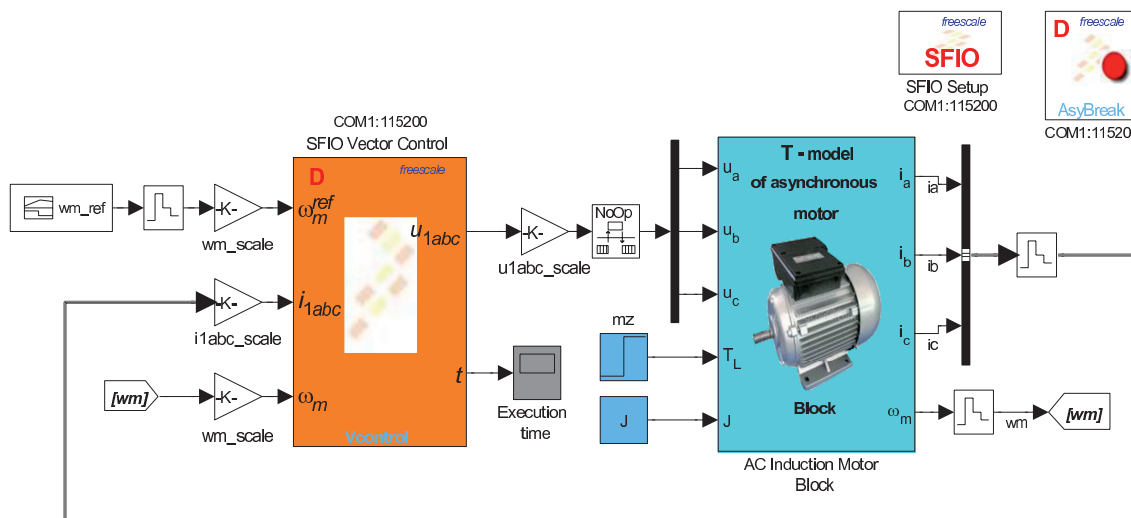
V našem případě bude veškerá řídicí část (regulátory, model pro výpočet magnetického toku, algoritmus odbuzování,...) přesunuta do DSC 56F8346, zatímco

model asynchronního motoru zůstává v prostředí MATLAB-Simulink. Situace je znázorněna na obrázku Obr. 12 a konkrétně na obrázku Obr. 13.

Komunikace je zajištěna pomocí SFIO Toolboxu, jež byl vyvinut na VUT FEKT ÚAMT [13]. Ten je rozdělen na dvě části. První část je na straně PC. Je realizovaná jako knihovna v MATLAB-Simulink. Obsažené bloky a funkce se starají především o zajištění komunikace přes sériovou linku, převod datových typů a čekání při příjmu a odesílání dat. Rovněž je k dispozici řada funkcí, jež použití knihovny usnadňují. Druhá část se nachází v cílovém zařízení. Stará se především o komunikaci. Je však také možno měřit výpočetní čas či vytvářet asynchronní breakpointy pro účely debugování.



Obr. 12: Znázornění metody Processor-in-the-loop pomocí SFIO Toolboxu

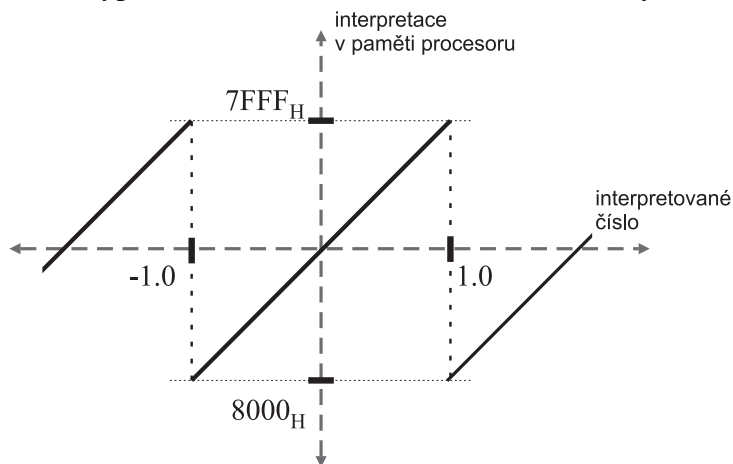


Obr. 13: Výsledné schéma simulace vektorového řízení s využitím metody Processor-in-the-loop v MATLAB-Simulink

## 4.2 Normalizace rovnic vektorového řízení pro výpočty v pevné řádové čárce

Vyjádření čísla v paměti počítače se obecně skládá ze znaménka, mantisi a exponentu, kde exponent rozhoduje o umístění desetinné čárky. Výpočetní jednotka s plovoucí čárkou je však poměrně složitá, což jednak může znamenat dlouhou dobu provedení instrukce a hlavně zvýšení ceny procesoru. Proto jsou u některých procesorů použity výpočty v pevné řádové čárce. Exponent je zde konstantou a není nutno jej uvádět, což spoří paměťové místo. Další výhodou je, že při sčítání a odčítání se s operandy zachází jako s celočíselnými typy. Exponent se obvykle volí tak, že celá část čísla je nulová. Pracuje se tak pouze s desetinnými čísly. Rozsah čísel v mikroprocesoru je tedy  $\langle -1,0; 1,0 \rangle$ . Výhodou je, že při násobení je výsledkem opět číslo v tomto rozsahu.

Hlavní nevýhodou výpočtů v pevné řádové čárce je možnost přetečení proměnné. Pro 16-bitové vyjádření čísla je rozsah  $\langle -1,0; 1,0 \rangle$  uložen jako  $\langle 8000_H; 7FFF_H \rangle$ , viz obrázek Obr. 14. Pokud k číslu  $7FFF_H$  přičteme jakoukoli kladnou hodnotu, dostaneme nesprávný výsledek. Při řízení to může znamenat fatální ztrátu kontroly. DSC 56F8346 je proto vybaven saturační logikou, která přetečení nedovolí. Stále to však znamená nekorektní provedení výpočtu a možnou následnou ztrátu kontroly.



**Obr. 14: Přetečení při počítání v pevné řádové čárce**

Aby se výsledky všech dílčích výpočtů nacházeli v rozsahu  $\langle -1,0; 1,0 \rangle$ , je třeba provést amplitudovou normalizaci. Ta byla používána například už v dobách operačních zesilovačů kvůli omezenému napětí na jejich výstupu. Základní myšlenkou je změna měřítko. Normalizaci obecné veličiny  $v$  lze provést jako

$$v = \left( \frac{v}{N_v} \right) N_v = v_n N_v, \quad (4.2.1)$$



kde  $v_n$  je bezrozměrná normalizovaná veličina, jejíž hodnota se nachází v rozsahu  $\langle -1,0;1,0 \rangle$  a  $N_v$  je normou, tedy maximální hodnotou o níž víme, že ji absolutní hodnota veličina  $v$  nepřekročí. Takto jsou přepsány všechny veličiny vystupující normalizované v rovnici. Ta je následně vyjádřena jako součet součinů veličin a příslušných konstant. Tyto konstanty však rovněž musí nabývat hodnoty v rozsahu  $\langle -1,0;1,0 \rangle$ . Pokud ne, je možné konstantu rozdělit na dílčí konstanty, jež tuto podmínku budou splňovat.

Aby byla zajištěna co nejvyšší přesnost výpočtu, doporučuje se, aby každá násobící konstanta nabývala absolutní hodnoty v intervalu  $\langle 0,5;1,0 \rangle$ . Ke konstantě tedy přibývá měřítko, což je celočíselná proměnná nesoucí počet násobení či dělení dvěma, respektive počet bitových posuvů doleva či doprava. Obecně například

$$v_n C_1 = v_n C_2 2^{-M_v}, \quad (4.2.2)$$

kde  $C_1$  je konstanta mimo interval  $\langle 0,5;1,0 \rangle$ ,  $C_2$  je konstanta v tomto intervalu a  $M_v$  je měřítko veličiny  $v$ . Jedná se o stejný princip jako u výpočtů v plovoucí čárce, kde je místo měřítka exponent.

V následujících kapitolách bude provedena normalizace všech rovnic použitých u vektorové řízení a algoritmů odbuzování.

## 4.2.1 Transformace souřadnicových systémů

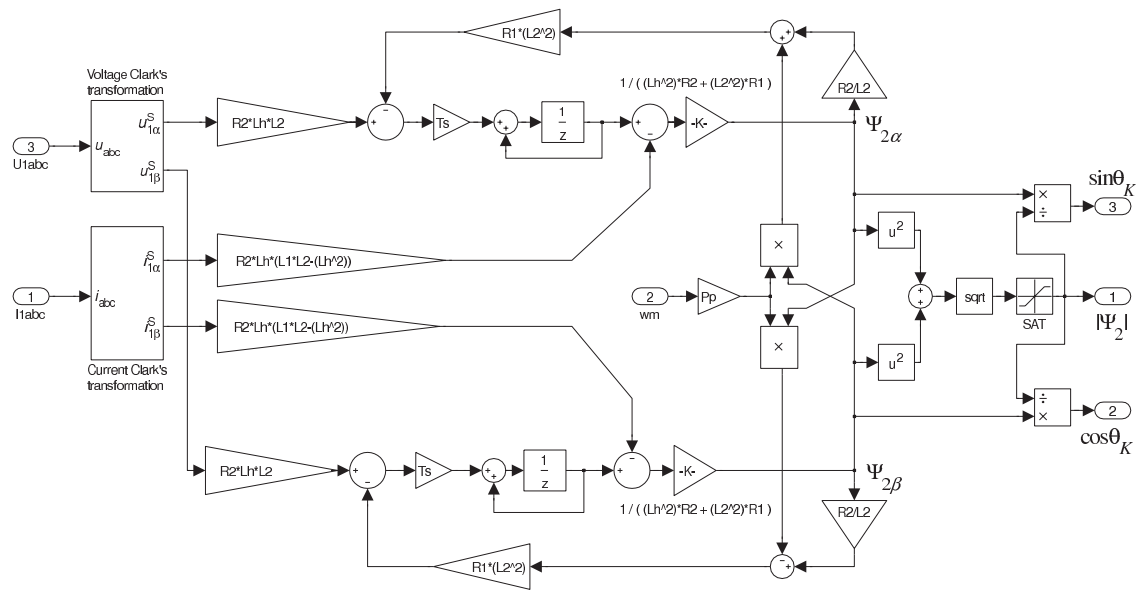
V této kapitole bude provedena normalizace přímé a zpětné Clarkové a Parkovy transformace. Tedy vztahů (1.1.3), (1.1.4), (1.2.2) a (1.2.3). Výhodou je, že se v rovnicích nachází veličiny stejných maximálních hodnot. Pokud pro ně zvolíme stejnou normu, dojde k jejímu vykrácení na obou stranách rovnic. Rovněž všechny násobící konstanty či hodnoty goniometrických funkcí se nacházejí v rozsahu  $\langle 0,5;1,0 \rangle$  Tedy krom konstanty  $2/\sqrt{3}$  v rovnici (1.1.3). Problém je odstraněn jejím rozdělením na dvě dílčí konstanty a provedení součtu ve správném pořadí. Pro přímou transformaci Clarkové tak získáváme

$$\begin{aligned} v_{1\alpha_n}(k) &= v_{1\alpha_n}(k) \\ v_{1\beta_n}(k) &= \left( \frac{1}{\sqrt{3}} v_{1\alpha_n}(k) + \frac{1}{\sqrt{3}} v_{1\beta_n}(k) \right) + \frac{1}{\sqrt{3}} v_{1\beta_n}(k), \end{aligned} \quad (4.2.3)$$

kde  $v$  je obecná veličina a  $k$  je pořadí vzorku. Rovnice (1.1.4), (1.2.2) a (1.2.3) lze použít beze změny, kdy namísto veličin vystupují jejich normalizované hodnoty.

## 4.2.2 Výpočet rotorového magnetického toku

Pro výpočet magnetického toku byl zvolen kombinovaný model dle rovnice (2.4.3). Pro jeho výpočet je potřeba znát mechanické otáčky rotoru, satorové napětí a proudy. V našem případě není satorové napětí měřeno, ale je uvažována jeho předchozí zadaná hodnota. Model je počítán ve statických  $\alpha$ - $\beta$  souřadnicích. Fázové napětí a proudy je tedy nutno do těchto souřadnic transformovat pomocí vztahu (4.2.3). Algoritmus výpočtu v diskrétním čase, implementovaný v prostředí MATLAB-Simulink, je na obrázku Obr. 15.



Obr. 15: Výpočet magnetického toku z modelu v prostředí MATALB-Simulink

Pokud si normalizované hodnoty v sumátorech označíme jako  $I_{\Psi\alpha}$  a  $I_{\Psi\beta}$ , pak normalizované složky magnetického toku lze získat ze vztahů

$$\Psi_{2\alpha_n}(k) = I_{\Psi\alpha}(k-1) - A \frac{N_{i_1}}{N_{\Psi_2}} i_{\alpha_n}(k) \quad (4.2.4)$$

$$\Psi_{2\beta_n}(k) = I_{\Psi\beta}(k-1) - A \frac{N_{i_1}}{N_{\Psi_2}} i_{\beta_n}(k),$$

kde

$$A = \frac{R_2 L_h (L_1 L_2 - L_h^2)}{L_h^2 R_2 + L_2^2 R_1}. \quad (4.2.5)$$

Nové hodnoty stavů v sumátorech pak získáme podle vztahů

$$\begin{aligned} I_{\Psi\alpha}(k) &= I_{\Psi\alpha}(k-1) + Bu_{1\alpha_n}(k-1) - C\omega_{m_n}(k)\Psi_{2\beta_n}(k) - D\Psi_{2\alpha_n}(k) \\ I_{\Psi\beta}(k) &= I_{\Psi\beta}(k-1) + Bu_{1\beta_n}(k-1) + C\omega_{m_n}(k)\Psi_{2\alpha_n}(k) - D\Psi_{2\beta_n}(k), \end{aligned} \quad (4.2.8)$$

kde

$$\begin{aligned} B &= T_s \frac{R_2 L_h L_2}{L_h^2 R_2 + L_2^2 R_1} \frac{N_{u1}}{N_{\Psi_2}} \\ C &= T_s P_p \frac{R_1 L_2^2}{L_h^2 R_2 + L_2^2 R_1} N_{\omega_m} \\ D &= T_s \frac{R_2}{L_2} \frac{R_1 L_2^2}{L_h^2 R_2 + L_2^2 R_1}. \end{aligned} \quad (4.2.7)$$

Datový typ pro uložení hodnot sumátorů se doporučuje volit 32-bitový (`Frac32` podle knihovny `mclib.h`), pro co nejvyšší přesnost. Ostatní proměnné mohou zůstat 16-bitové.

Výsledná normalizovaná hodnota rotorového magnetického toku je získána vztahem

$$|\Psi_{2_n}(k)| = \sqrt{\Psi_{2\alpha_n}^2(k) + \Psi_{2\beta_n}^2(k)}, \quad (4.2.8)$$

kde odmocnina je realizována využitím funkce `MCLIB_Sqrt()` z knihovny `mclib.h`. Normu magnetického toku je třeba volit s dostatečnou rezervou, aby při součtu ve vztahu (4.2.8) nemohlo dojít k přetečení. To hrozí zejména po zapnutí motoru při vytváření požadované hodnoty magnetického toku.

Jak už bylo uvedeno v kapitole 1.2, konkrétně ve vztazích (1.2.4) a (1.2.5), je pro účely Parkovy transformace vhodné počítat přímo  $\sin\theta_K$  a  $\cos\theta_K$ . Tedy pomocí normalizovaných vztahů

$$\begin{aligned} \sin\theta_K &= \frac{\Psi_{2\beta_n}(k)}{|\Psi_{2_n}(k)|} \\ \cos\theta_K &= \frac{\Psi_{2\alpha_n}(k)}{|\Psi_{2_n}(k)|}. \end{aligned} \quad (4.2.9)$$

Aby nedošlo k přetečení, musí být jmenovatel vždy nenulový a větší než číselník. Aby obojí platilo i po spuštění algoritmu, je nutno omezit minimální hodnotu  $\Psi_{2_n}$  na malou zvolenou hodnotu (blok SAT na obrázku Obr. 15).

### 4.2.3 Výpočet úhlové rychlosti magnetického toku

Výpočet je prováděn na základě vztahu (2.4.4). Pro nízké otáčky, kdy je proud  $i_{1d}$  jako jmenovatel příliš malý, je úhlová rychlost počítána jako

$$\omega_f = P_p \omega_m. \quad (4.2.10)$$

Po provedení normalizace vztahu (4.2.10) dostáváme

$$\omega_{f_n} = P_p \frac{N_{\omega_m}}{N_{\omega_f}} \omega_{m_n}. \quad (4.2.11)$$

Pokud v předchozím vztahu vhodně zvolíme normy úhlových rychlostí, bude mezi jejich normalizovanými hodnotami rovnost.

Po dosažení dostatečné velikosti proudu  $i_{1d}$  lze normalizovanou hodnotu úhlové rychlosti pole získat ze vztahu

$$\omega_{f_n} = \frac{R_2}{L_2} \frac{1}{N_{\omega_f}} \frac{i_{1q_n}}{i_{1d_n}} + P_p \frac{N_{\omega_m}}{N_{\omega_f}} \omega_m. \quad (4.2.12)$$

### 4.2.4 Normalizace diskrétního PI a PID regulátoru

Přenosy PI a PID regulátorů byly uvedeny v kapitole 2.3. Jejich konkrétní realizace v prostředí MATLAB-Simulink je na obrázcích Obr. 16 a Obr. 17.

Pokud hodnotu v sumátoru diskrétního PI regulátoru označíme jako  $I_{PI}$ , pak jeho normalizovaný, nelimitovaný akční zásah  $x_n$  vypočteme pomocí vztahu

$$x_n(k) = K_{PI_n} w_n(k) - K_{PI_n} y_n(k) + I_{PI_n}(k-1), \quad (4.2.13)$$

kde  $w_n$  je normalizovaná žádaná hodnota,  $y_n$  je normalizovaná skutečná hodnota a

$$K_{PI_n} = K_{PI} \frac{N_{w,y}}{N_x} \quad (4.2.14)$$

je normalizované zesílení. Aby mohlo dojít ke korektnímu výpočtu odchylky na vstupu regulátoru, předpokládá se  $N_{w,y}=N_w=N_y$ .

Nová hodnota v sumátoru pak odpovídá výsledku normalizovaného vztahu

$$I_{PI_n}(k) = I_{PI_n}(k-1) + K_{PI_n} w_n(k) - K_{PI_n} y_n(k) + T_s T_i (x_n^{max}(k) - x_n(k)), \quad (4.2.15)$$

kde  $x_n^{max}$  je normalizovaný, limitovaný akční zásah a

$$K_{PI_n} = \frac{K_{PI} T_s}{T_i} \frac{N_{w,y}}{N_x} \quad (4.2.16)$$

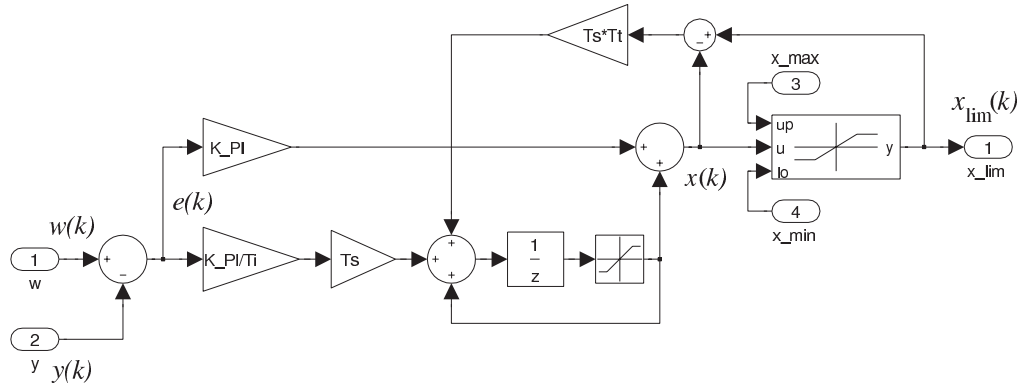
je normalizované integrační zesílení. Datový typ uchovávající hodnotu v sumátoru je, z důvodu vyšší přesnosti, 32-bitový. Do sumátoru v modelu na obrázku Obr. 16 byla přidána saturace na hodnotu  $N_x$ , tak aby situace lépe odpovídala poměrům v DSC. To platí i pro sumátory v PID regulátoru na obrázku Obr. 17.

Jak je vidět ve vztahu (4.2.15), pokud by bylo při výpočtu použito saturační logiky, nebylo by možno použít zvolenou metodu anti-windupu, neboť by v případě  $x_n^{max}=1,0$  nikdy nemohlo platit  $|x_n| > |x_n^{max}|$ . Rovněž pokud by zesílení integrační složky ve vztahu (4.2.16) přesahovalo hodnotu 0,5, mohlo by docházet k saturaci odchylky a ke zbytečnému zpomalení „odintegrování“ hodnoty v sumátoru. Proto saturační logika nebyla použita a bylo využito čtyř záložních bitů v horní části 36-bitových akumulátorů. V případě přetečení 32-bitového rámce v nich totiž může být uložena celá část čísla až do hodnoty menší než 16,0 (znaménkový bit existuje zvlášť). To však znemožňuje využití jazyka C, neboť překladač používá instrukce jež práci s těmito bity neumožňují. Regulátory tak byly napsány jako funkce v assembleru. Na jednu stranu to znamená omezení přenositelnosti kódu na jiné procesory, na druhou stranu však mohl být dobře optimalizován. A to například využitím duální Harvardské architektury, respektive paralelního čtení dvou operandů.

Při návrhu je však nutno dodržet některá pravidla pro hodnoty konstant, jinak může snadno dojít k přetečení některé z proměnných a k případné ztrátě kontroly. Uvažujme nejhorší možný případ. Odchylka  $|e_n(k)| = |w_n(k) - y_n(k)|$  může teoreticky dosáhnout až hodnoty téměř 2,0. Pak v případě, že hodnota v integrátoru může nabývat maximálně absolutní hodnoty 1,0, musí být hodnota zesílení  $K_{PI_n}$  menší než 7,5. Pro konstantu anti-windupu  $T_i$  je vhodné, aby násobek  $T_s T_i$  byl v měřítku alespoň 4. Pak je zajištěno  $T_s T_i (x_n^{max} - x_n) < 1,0$ . Normalizované integrační zesílení pak může nabývat hodnoty maximálně 7,0. Po výpočtu rovnice (4.2.15) je její výsledek saturován tak, aby jej bylo možno uložit mimo akumulátor do 32-bitové proměnné. Pokud budou dodrženy všechny tyto podmínky, je správná funkce regulátoru zajištěna.

Je-li nezbytně nutné, aby byla některá z konstant větší než maximální hodnota, je možno využít celkového měřítka. To znamená, že měřítka všech konstant (krom  $T_s T_i$ ), včetně normalizovaných hodnot omezení  $x_n^{max}$ , jsou zvětšena o hodnotu celkového

měřítka tak, aby byly splněny všechny uvedené podmínky. Akční zásah je pak bitově posunut doleva o hodnotu celkového měřítka. Ač jsou hodnoty všech veličin uvnitř regulátoru v menším měřítku, na výstupu se to neprojeví. Avšak dochází k částečné ztrátě přesnosti. Tuto možnost je tedy doporučeno používat až v krajních případech.



**Obr. 16: Realizace použitého paralelního PI regulátoru**

Pokud hodnotu v sumátoru integrační složky diskrétního PID regulátoru označíme jako  $I_{PID}$  a hodnotu sumátoru filtrované derivační složky jako  $D_{PID}$ , pak nelimitovaný, normalizovaný akční zásah získáme z rovnice

$$x_n(k) = K_{PID_n} w_n(k) - K_{PID_n} y_n(k) + I_{PID_n}(k-1) - AND_{PID_n}(k-1), \quad (4.2.17)$$

kde

$$K_{PID_n} = (N+1)K_{PID} \frac{N_{w,y}}{N_x} \quad (4.2.18)$$

je normalizované zesílení PID regulátoru a

$$AN = \left( e^{\frac{T_s \cdot N}{T_d}} - 1 \right) N, \quad (4.2.19)$$

kde  $N$  je koeficient filtrace derivační složky.

Novou hodnotu sumátoru v integrační složce lze získat podle vztahu

$$I_{PID_n}(k) = I_{PID_n}(k-1) + K_{PID_n} w_n(k) - K_{PID_n} y_n(k) + T_s T_t (x_n^{lim}(k) - x_n(k)). \quad (4.2.20)$$

V předchozím vztahu pak platí

$$K_{IPID_n} = K_{IPID} \frac{T_s}{T_i} \frac{N_{w,y}}{N_x}, \quad (4.2.21)$$

což je zesílení integrační složky PID regulátoru.

Hodnota sumátoru ve filtrované derivační složce je počítána podle rovnice

$$D_{PID_n}(k) = AD_{PID_n}(k-1) + K_{DPID_n} w_n(k) - K_{DPID_n} y_n(k), \quad (4.2.22)$$

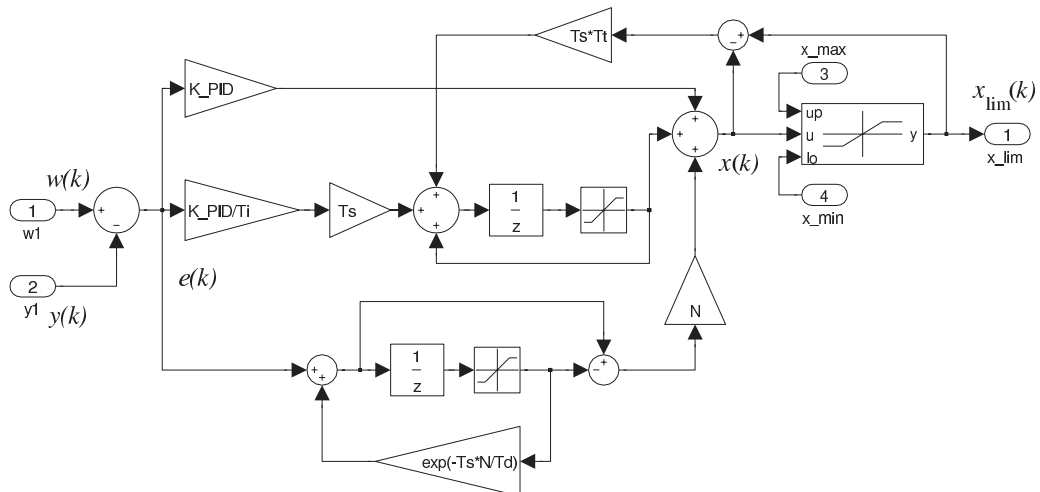
kde

$$A = e^{-\frac{T_s \cdot N}{T_d}}, \quad (4.2.23)$$

$$K_{DPID_n} = K_{DPID_n} \frac{N_{w,y}}{N_x}$$

je normalizované zesílení derivační složky. Hodnoty obou sumátorů jsou po výpočtu satureovány a uloženy ve 32-bitových proměnných.

Stejně jako u PI regulátoru, i zde jsou výpočty prováděny bez saturační logiky. Opět je tedy nutno dodržet jistá pravidla návrhu tak, aby žádný z výsledků nepřekročil hodnotu 16,0. Pokud je hodnota konstanty  $AN$  menší než jedna (platí pro malá  $N$  a větší  $T_d$ ), může zesílení ve vztahu (4.2.18) nabývat hodnoty až 7,0. Pro konstanty ve vztahu (4.2.20) platí řečené u PI regulátoru. U rovnice (4.2.22) pak neplatí žádná zvláštní pravidla, neboť ve vztahu (4.2.23) je vždy  $A < 1$  a navíc platí  $K_{PID_n} = (N+1)K_{DPID_n}$ .



Obr. 17: Realizace použitého paralelního PID regulátoru

## 4.3 Úprava vybraných algoritmů odbuzování pro výpočty v pevné řádové čárce

V této podkapitole budou normalizovány všechny rovnice použité v algoritmech odbuzování uvedených v kapitole 3.

### 4.3.1 Algoritmus $1/\omega_f$

Jak bylo uvedeno v kapitole 3.1, výpočet žádané hodnoty toku je založen na jediné rovnici (3.1.1). Normalizovanou hodnotu reference toku pak získáme ze vztahu

$$\Psi_{2d_n}^{ref}(k) = \frac{\omega_{base_n} \Psi_{2d_n}^{max}}{|\omega_{f_n}(k-1)|}, \quad (4.3.1)$$

kde

$$\omega_{base_n} \Psi_{2d_n}^{max} = \frac{k_{1/\omega_f} L_h U_1^{max} \Psi_{2d}^{max}}{L_1 \sqrt{(\Psi_{2d}^{max})^2 (1-\sigma^2) + \sigma^2 L_h^2 (I_1^{max})^2}} \frac{1}{N_{\omega_f} N_{\Psi_2}}. \quad (4.3.2)$$

Konstanta definovaná rovnicí (4.3.2) byla odvozena dle vztahu (3.1.4). Při dělení je nutno zajistit (například podmínkou), aby vždy platilo že čitatel je vždy menší než jmenovatel. Jinak dojde k přetečení, i v případě požití saturační logiky (dělení není saturační logikou ovlivněno). Výsledek rovnice (4.3.1) je třeba limitovat do námi požadovaného rozsahu.

### 4.3.2 Odbuzování s PI regulátorem

Základem algoritmu je diskretní PI regulátor popsáný v kapitole 4.2.4. Jak bylo uvedeno v kapitole 3.2, pro odstranění odmocniny je počítáno s druhými mocninami  $d$ - $q$  napěťových složek. Normalizovanou regulovanou veličinu a normalizovanou žádanou hodnotu na vstupu PI regulátoru získáme ze vztahů

$$\begin{aligned} y_n(k) &= u_{1d_n}^2(k-1) + u_{1q_n}^2(k-1) \\ w_n(k) &= k_{PI}^2, \end{aligned} \quad (4.3.3)$$

kde  $k_{PI}$  je volitelný koeficient (viz kapitola 3.2) a společná norma je  $N_{u_l}^2$ . Aby při sčítání ve vztahu (4.3.3) nedocházelo k saturaci, je nutno volit napěťovou normu s dostatečnou rezervou. Obvykle to však nebývá problémem.



### 4.3.3 Odbuzování na maximální moment

Funkce algoritmu byla probrána v kapitole 3.3. Opět jsou použity diskrétní PI regulátory popsané v kapitole 4.2.4. Je tedy třeba dbát uvedených požadavků na jejich návrh. Pro žádanou a regulovanou veličinu PI regulátoru reference magnetického toku platí rovnice (4.3.3). Normalizovaná regulovaná veličina a normalizovaná žádaná hodnota PI regulátoru proudového omezení  $i_{1q}^{max}$  jsou získány vztahy

$$\begin{aligned} y_n(k) &= \left| u_{1d_n}(k-1) \right| \\ w_n(k) &= \frac{1}{\sqrt{2}}, \end{aligned} \quad (4.3.4)$$

kde společnou normou je  $N_{u_1}$ .

### 4.3.4 Algebraické odbuzování na maximální moment

Jak je v kapitole 3.4 uvedeno, celá oblast odbuzování lze, z hlediska rychlosti  $\omega_f$ , rozdělit na dva regiony. Normalizovanou rychlost přechodu do prvního regionu  $\omega_{base1}$  lze určit z rovnice

$$\omega_{base1_n} = \frac{\sqrt{\left(k_{Algeb} U_1^{max} L_1 I_d\right)^2 + \left(R_1 I_q\right)^2} - R_1 I_q}{L_1 I_d} \frac{1}{N_{\omega_f}}. \quad (4.3.5)$$

Omezení proudu  $i_{1q}$  je v prvním regionu neovlivněno, respektive je počítán dle vztahu (2.3.4), jehož normalizovanou obdobou je

$$i_{1q_n}^{max} = \sqrt{\left(I_1^{max}\right)^2 - \left(i_{1d_n}^{ref}\right)^2}. \quad (4.3.6)$$

Referenci magnetického toku lze získat ze vztahu (3.4.4) násobeného hlavní indukčností  $L_h$ . Pro normalizaci je však třeba rovnici upravit. Dostáváme

$$\Psi_{2d}^{ref} = L_h \sqrt{\frac{\left(\frac{k_{Algeb} U_1^{max}}{\omega_f}\right)^2 - \left(\sigma L_1 I_1^{max}\right)^2}{L_1^2 - \sigma^2 L_1^2}} = \sqrt{\frac{\left(L_h k_{Algeb} U_1^{max}\right)^2 - \left(L_h \sigma L_1 I_1^{max}\right)^2}{L_1^2 - \sigma^2 L_1^2} \omega_f^2}. \quad (4.3.7)$$

Po provedení normalizace pak platí

$$\Psi_{2d_n}^{ref}(k) = \sqrt{\frac{A - B\omega_{f_n}^2(k-1)}{\omega_{f_n}^2(k-1)}}, \quad (4.3.8)$$

kde

$$A = \frac{(L_h k_{Algeb} U_1^{max})^2}{L_1^2 - \sigma^2 L_1^2} \frac{1}{N_{\Psi_{2d}}^2 N_{\omega_f}^2} \quad (4.3.9)$$

$$B = \frac{(L_h \sigma L_1 I_1^{max})^2}{L_1^2 - \sigma^2 L_1^2} \frac{1}{N_{\Psi_{2d}}^2}.$$

Aby dělení proběhlo korektně je nutné, aby byl jmenovatel vždy větší než čítec. Zároveň můžeme požadovat, aby k dělení docházelo až v případě nutnosti, tedy až v případech, kdy pro výsledek rovnice (4.3.8) platí  $\Psi_{2d_n}^{ref} < \Psi_{2d_n}^{max}$ . Pak musí platit

$$|\omega_{f_n}| > \sqrt{\frac{A}{B + (\Psi_{2d_n}^{max})^2}}. \quad (4.3.10)$$

Odmocnina ve vztazích (4.3.6) a (4.3.8) je realizována pomocí funkce `MCLIB_Sqrt()`, z knihovny `mclib.h`.

Do druhého regionu odbuzování se přechází po překročení úhlové rychlosti pole, definované vztahem (3.4.5). Jeho normalizovanou hodnotu získáme jako

$$\omega_{base2} = \sqrt{\frac{1 + \sigma^2}{2\sigma^2 L_1^2}} \frac{k_{Algeb} U_1^{max}}{I_1^{max}} \frac{1}{N_{\omega_f}}. \quad (4.3.11)$$

Reference magnetického toku je pak určena pomocí rovnice (3.4.6). Její normalizací je získán vztah

$$\Psi_{2d_n}^{ref}(k) = \frac{C}{\omega_{f_n}(k-1)} = \frac{k_{Algeb} U_1^{max}}{\sqrt{2} L_1} \frac{1}{N_{\Psi_{2d}} N_{\omega_f}} \frac{1}{\omega_{f_n}(k-1)}. \quad (4.3.12)$$

Maximální hodnota  $q$ -složky proudu je počítána z normalizovaného vztahu

$$i_{1q_n}^{max}(k) = \frac{D}{\omega_{f_n}(k-1)} = \frac{k_{Algeb} U_1^{max}}{\sqrt{2}\sigma L_1} \frac{1}{N_{i_1} N_{\omega_f}} \frac{1}{\omega_{f_n}(k-1)}. \quad (4.3.13)$$

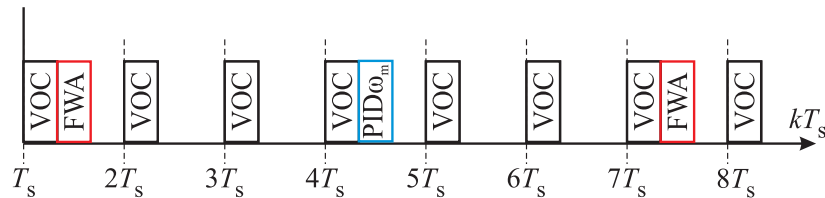
Opět zde musí platit, že čitatel bude vždy menší než jmenovatel. Někdy to může znamenat, že konstanta  $C$  a  $D$  nebude v rozsahu  $(0,5; 1,0)$ .

## 4.4 Výsledný algoritmus

Vývojový diagram hlavní smyčky výsledného algoritmu vektorového řízení asynchronního motoru je v příloze B.

Bloky „čtení vstupu“ a „zápis na výstup“ je myšlena komunikace SFIO Toolboxu s modelem asynchronního motoru v MATLAB-Simulink, respektive čtení a zápis do příslušných datových struktur.

Rovněž bylo znázorněno počítání algoritmu odbuzování a diskrétního PID regulátoru  $\omega_m$  se vzorkovací periodou delší než zbytek algoritmu. Proměnná  $kT_s$  zde nese relativní pořadí vzorku. Pokud je její hodnota násobkem  $n$ , dojde k výpočtu příslušného algoritmu. Aby bylo dosaženo úspory výpočetního času, nejsou PID regulátor a algoritmus odbuzování počítány ve stejné iteraci hlavní smyčky programu. Lépe je situace znázorněna v časovém diagramu na následujícím obrázku.



VOC=Vector oriented control (vektorové řízení)

FWA=Field weakening algorithm (algoritmus odbuzování)

PID $\omega_m$ =Algoritmus výpočtu PID regulátoru

**Obr. 18: Jednoduchý časový diagram řazení výpočtů pro  $n=4$**

Výpočet nelineárních částí  $d$ - $q$  složek napětí je proveden podle vztahů (2.2.5), respektive jejich normalizovaných analogií

$$u_{1d_n}^{nlin} = -\sigma L_1 \frac{N_{\omega_f} N_{i_1}}{N_{u_1}} \omega_{f_n} i_{1q_n}$$

$$u_{1q_n}^{nlin} = L_1 \frac{N_{\omega_f} N_{i_1}}{N_{u_1}} \omega_{f_n} i_{1d_n}. \quad (4.4.1)$$

Napětí  $u_{1d}$  je vypočteno dle vztahu (2.2.3), tedy jako součet akčního zásahu PI regulátoru  $i_{1d}$  a nelineární  $d$ -složky napětí.

Omezení  $q$ -složky napětí je určeno dle normalizovaného vztahu (2.3.2), tedy

$$u_{1q_n}^{max} = \sqrt{(U_1^{max})^2 - u_{1d_n}^2}. \quad (4.4.2)$$

Výsledné napětí  $u_{1q}$  je pak vypočteno jako součet akčního zásahu regulátoru  $i_{1q}$  a nelineární  $q$ -složky napětí. Následně je omezeno výsledkem vztahu (4.4.2). Transformace souřadnic byly popsány v kapitole 1.1, 1.2 a 4.2.1.

## 5 SIMULACE ALGORITMŮ ODBUZOVÁNÍ

V této kapitole budou srovnány výše popsané algoritmy odbuzování na základě simulací v MATLAB-Simulink verze R2008b ve spolupráci s DSC 56F8346.

Jak bylo uvedeno v kapitole 3, lze oblast odbuzování rozdělit na dva regiony. Ne u všech asynchronních motorů je však možno druhého regionu dosáhnout. Obecně jej podle vztahu (3.4.5) mohou dosáhnout pouze stroje s velkým činitelem rozptylu  $\sigma$  a velkým proudovým omezením. Aby bylo možno srovnat výsledky algoritmů odbuzování pro oba regiony, budou provedeny simulace vektorového řízení dvou asynchronních motorů.

Vzhledem k tomu, že v oblasti kde dochází k odbuzování se v modelu nachází řada nelinearit, byla optimalizace algoritmů provedena pomocí Nelder-Meadovy metody. Ta využívá simplexu, tedy geometrického útvaru, jež má ve dvou rozměrech tvar rovnostranného trojúhelníka a pravidelného jehlanu ve třech rozměrech (počet krajních bodů je vždy o jednu vyšší než dimenze prostoru). Ve všech krajních bodech jsou vypočteny hodnoty minimalizační funkce. Tu definuje uživatel. Bod, v němž byla zjištěna její nejvyšší hodnota je zrcadlen podle zbylých bodů, jež tvoří osu souměrnosti. Během výpočtu může být délka strany změněna. Stejně tak zrcadlený bod nemusí být umístěn do stejné vzdálenosti od osy souměrnosti. Takto se pokračuje, dokud není v určitých mezích nalezeno lokální minimum [14]. Pro realizaci metody byla využita funkce `fminsearch()` v programu MATLAB. Jako minimalizační funkce bylo zvoleno ITAE kritérium. Jeho diskretní verze je v našem případě definována vztahem

$$ITAE = \sum_{k=\frac{T_0}{T_s}}^{\frac{T_1}{T_s}} \left| \omega_m^{ref}(k) - \omega_m(k) \right| (kT_s - T_0), \quad (5.1)$$

kde  $T_0$  je čas (násobek  $T_s$ ) spuštění,  $T_1$  je čas ukončení výpočtu kritéria a  $k$  je číslo vzorku. Optimalizovanou veličinou je obvykle parametr  $k_x$ , kde  $x$  značí použitý algoritmus odbuzování (viz kapitola 3), ale také nastavení PI regulátorů.

Hlavními aspekty srovnání algoritmů bude velikost dosaženého momentu, což se projeví v hodnotě ITAE a výpočetní náročnost algoritmu. Doba výpočtu algoritmu je měřena pomocí příslušných funkcí obsažených v SFIO Toolboxu.

## 5.1 Simulace motoru s dominantním prvním regionem

Hodnoty pro matematický model asynchronního motoru, včetně dalších vstupních parametrů simulace jsou uvedeny v tabulce Tab. 1. Jedná se o 2,2 kW, dvoupólový motor [15]. Proudový limit byl zvolen asi 1,3 násobek jmenovitého. To, že nelze druhého regionu dosáhnout, lze ověřit výpočtem úhlové rychlosti  $\omega_{base2}$ , podle vztahu (3.4.5). Dostáváme

$$\omega_{base2} = \sqrt{\frac{1+\sigma^2}{2\sigma^2 L_1^2} \frac{U_1^{max}}{I_1^{max}}} = \sqrt{\frac{1+0,072^2}{2 \cdot 0,072^2 \cdot 0,348^2} \frac{250}{10}} = \underline{\underline{711,7 \text{ rad/s}}},$$

což je pro dvoupólový motor velmi vysoká hodnota.

Velikosti norem použitých při implementaci do DSC 56F8346 jsou v příslušné tabulce, v příloze 3. Nastavení všech regulátorů, včetně regulátoru použitého v algoritmu odbuzování s PI regulátorem, jsou v příloze 4. Proudové regulátory byly nastaveny dle pravidel uvedených v kapitole 2.3 a 4.2.4. Časová konstanta přenosu řízení proudu  $i_{1d}$  byla zvolena na 2,5 ms a u proudu  $i_{1q}$  na 250  $\mu$ s. Tedy tak, aby byly proudové smyčky co nejrychlejší, při zachování vzorkovacího teoremu a pravidel v kapitole 4.2.4. Počáteční návrh PID regulátorů byl proveden pomocí nástroje sisotool v prostředí MATLAB na požadovaný tvar skokové odezvy. Avšak díky nelinearitám (nekonstantní magnetický tok, limitace akčních zásahů) a také kvůli některým omezením při implementaci do DSC, bylo nutno hodnoty upravit ručně.

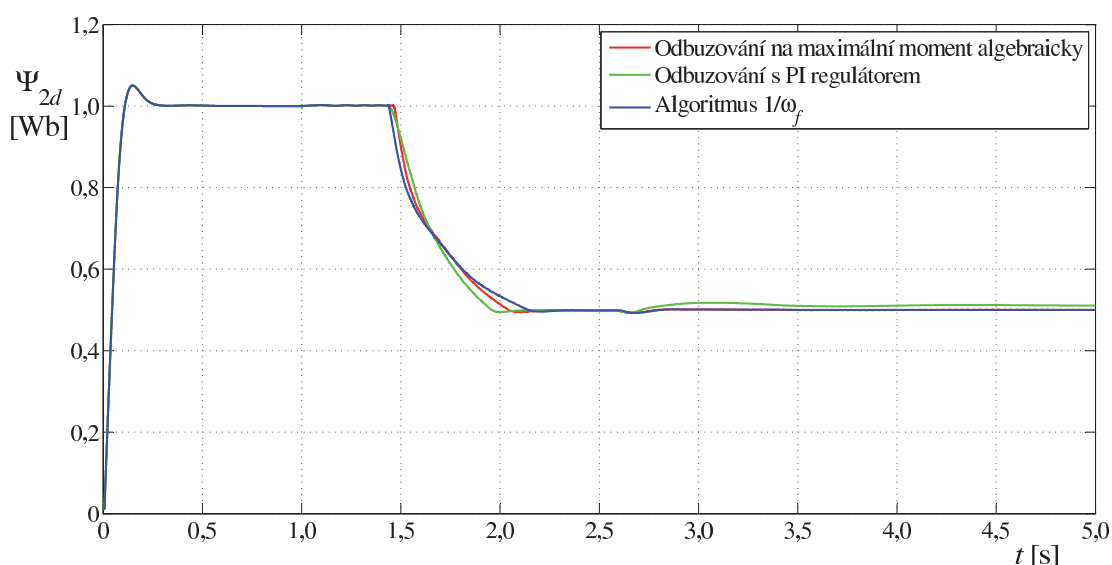
Veličina	Hodnota	Jednotka
$I_1^{max}$	10	A
$J$	0,03	kgm <sup>2</sup>
$k_{1/\omega f}$	0,915	-
$k_{Algeb}$	0,959	-
$k_{PI}$	0,950	-
$L_1$	0,3480	H
$L_2$	0,3515	H
$L_h$	0,3370	H
$m_z$	0	Nm
$P_p$	1	-
$R_1$	2,01	$\Omega$
$R_2$	1,78	$\Omega$
$T_s$	120	$\mu$ s
$T_{s2}$	960	$\mu$ s
$T_{sim}$	12	$\mu$ s
$U_1^{max}$	250	V
$\sigma$	0,072	-
$\Psi_{2d}^{max}$	1,0	Wb
$\Psi_{2d}^{min}$	0,5	Wb

Tab. 1: Vstupní parametry simulace motoru s dominantním prvním regionem odbuzování

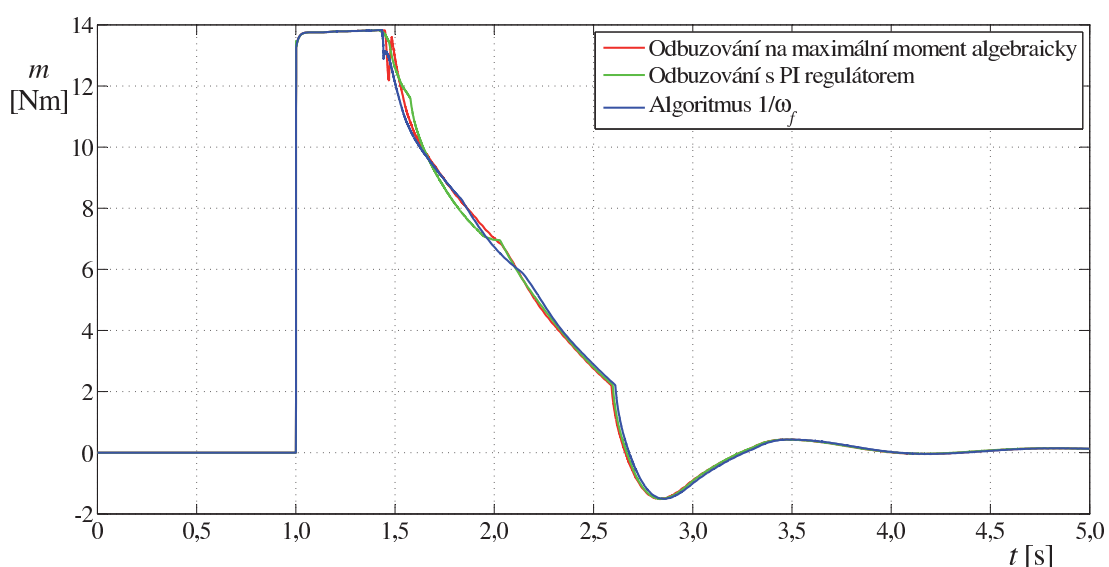
Vzorkovací perioda byla zvolena 120  $\mu\text{s}$ . U algoritmů odbuzování pak jako osminásobek. Simulace probíhala s pevným časovým krokem  $T_{sim}=12 \mu\text{s}$ .

U algoritmů odbuzování  $1/\omega_f$  a algebraického odbuzování byl optimalizován parametr  $k_x$ . U algoritmu odbuzování s PI regulátorem bylo  $k_{PI}$  zvoleno a následně optimalizováno jeho zesílení a integrační časová konstanta.

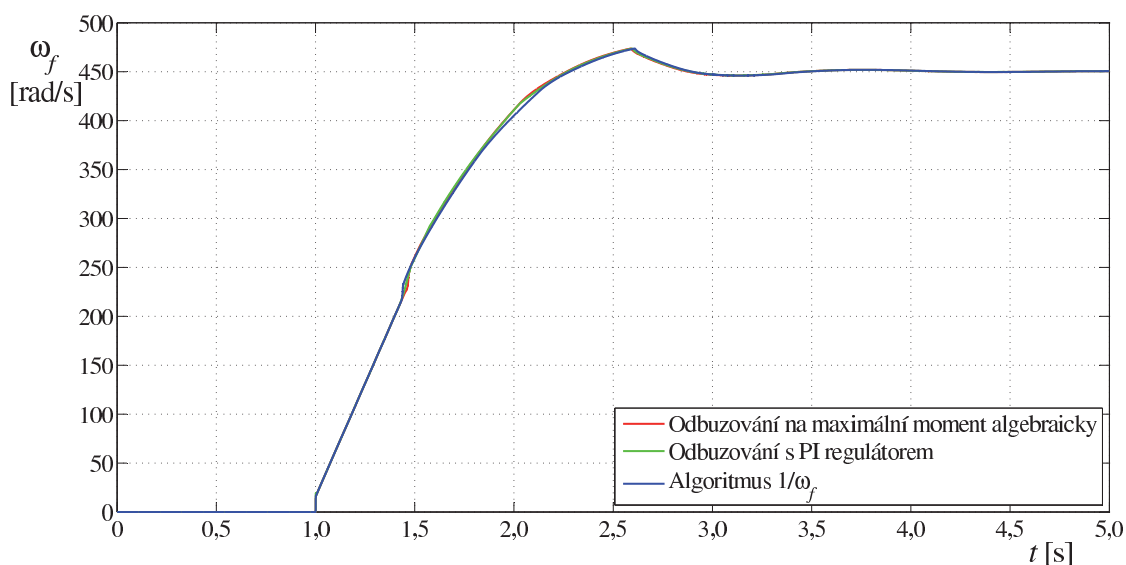
Před požadavkem na změnu otáček byla ponechána doba jedné sekundy pro ustálení magnetického toku. Motor byl následně roztočen na 450 rad/s bez zátěžného momentu. Vzhledem k tomu, že odbuzování s PI regulátorem a na maximální moment jsou v prvním regionu totožné, nebylo odbuzování na maximální moment simulováno.



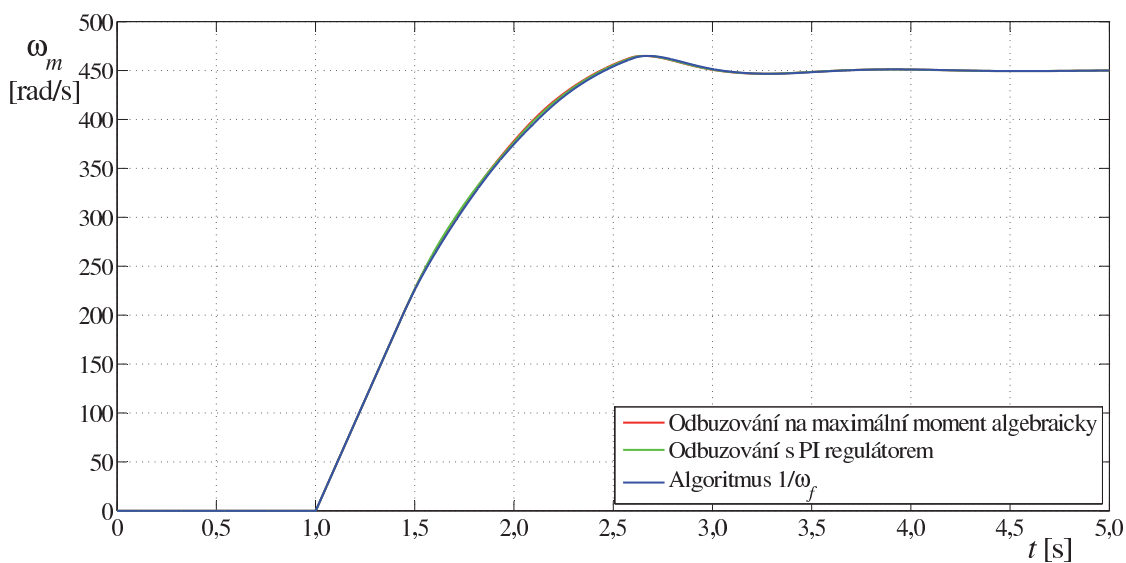
Obr. 19: Časová závislost magnetického toku u motoru s dominantním prvním regionem



Obr. 20: Časová závislost momentu u motoru s dominantním prvním regionem



**Obr. 21: Časová závislost úhlové rychlosti  $\omega_f$  u motoru s dominantním prvním regionem**



**Obr. 22: Časová závislost mechanické úhlové rychlosti u motoru s dominantním prvním regionem**

Jak je vidět z grafů na obrázcích Obr. 19 až Obr. 22, všechny algoritmy odbuzování podávají velmi podobné výsledky. Srovnání dosažených hodnot  $ITAE$  kritéria, měřeného v čase 1 s až 5 s, včetně relativního srovnání počítaného dle vztahu

$$\delta_{ITAE} = 100 \frac{ITAE - ITAE_0}{ITAE}, \quad (5.1.1)$$

kde  $ITAE_0$  je nejlepší a  $ITAE$  srovnávaný výsledek, je v tabulce Tab. 2. Teoreticky nejlepších výsledků v prvním regionu lze dosáhnout s odbuzováním na maximální



moment algebraicky. Jedná se však o zanedbatelné rozdíly, maximálně v řádech procent.

Algoritmus odbuzování	ITAE [rad]	$\delta_{ITAE}$ [%]
na maximální moment algebraicky	115,89	0,00
s PI regulátorem	116,47	0,49
$1/\omega_f$	118,65	2,39

**Tab. 2: Hodnoty ITAE kritéria dosažených algoritmy odbuzování při simulaci motoru s dominantním prvním regionem**

Zákmit momentu u algoritmu odbuzování na maximální moment, v grafu na obrázku Obr. 20, je způsoben přechodem do prvního regionu odbuzování, respektive nenávazností příslušných vztahů v tomto bodě.

Zlom v závislostech  $\omega_m(t)$  a  $\omega_f(t)$ , v čase 2,6 s, je způsoben tím, že proudové omezení dané vztahem (2.3.4) dává vyšší hodnoty, než je schopen regulátor proudu  $i_{1q}$  vzhledem k napěťovým omezením dosáhnout. Chvilí tak trvá, než se integrační složka PID regulátoru  $\omega_m$  odintegruje.

## 5.2 Simulace pro motor s dosažitelným druhým regionem

Parametry simulace, včetně hodnot náhradního modelu asynchronního motoru, jsou uvedeny v tabulce Tab. 3. Tentokrát je simulován 22 kW, čtyřpólový asynchronní motor [9]. O dosažitelnosti druhého regionu se můžeme přesvědčit výpočtem vztahu (3.4.5), tedy

$$\omega_{base2} = \sqrt{\frac{1+\sigma^2}{2\sigma^2 L_1^2} \frac{U_1^{max}}{I_1^{max}}} = \sqrt{\frac{1+0,2245^2}{2 \cdot 0,2245^2 \cdot 0,0158^2} \frac{200}{80}} = \underline{\underline{535,21 \text{ rad/s.}}}$$

S ohledem na další parametry simulace se jedná o dosažitelnou hodnotu. Je možno si povšimnout, že činitel rozptylu  $\sigma$  a proudové omezení  $I_1^{max}$  jsou mnohem větší, než u motoru s dominantním prvním regionem (viz tabulka Tab. 1 a Tab. 3).

Zvolené hodnoty norem pro implementaci do DSC jsou uvedeny v příloze 3, v příslušné tabulce.

Nastavení všech regulátorů, včetně těch použitých u algoritmů odbuzování, jsou v příloze 4, v příslušné tabulce. Postup návrhu byl obdobný jako v předchozí kapitole 5.1.

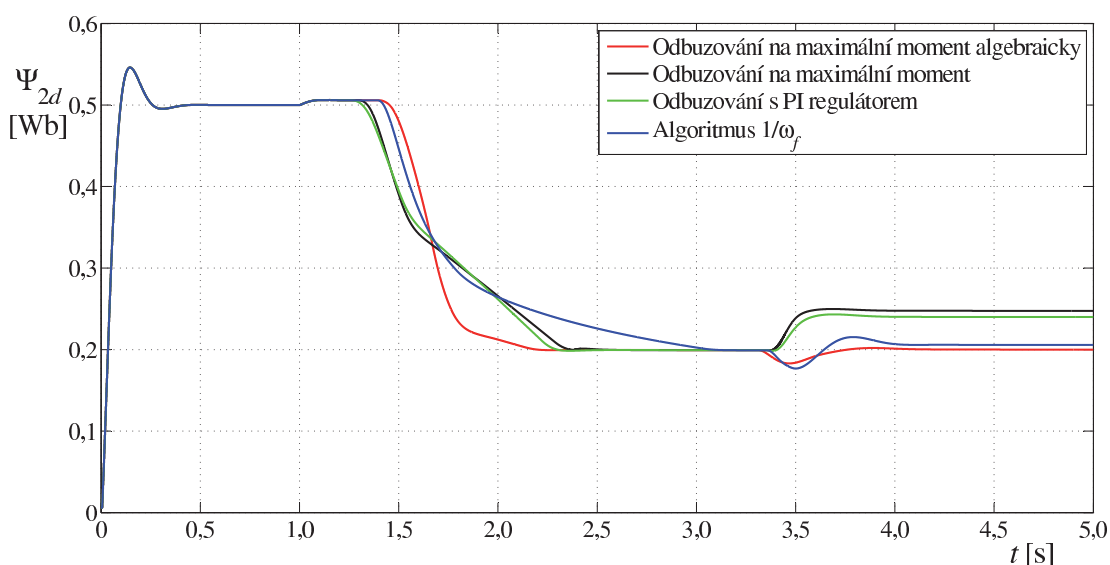
Vzorkovací perioda  $T_{s2}$  byla zvolena pouze čtyřnásobná oproti  $T_s$ , jelikož mezi algoritmem odbuzování a PID regulátorem  $\omega_m$  se nachází vazba přes limit proudu  $i_{1q}^{max}$ . Ostatní vzorkovací periody zůstaly stejné jako v kapitole 5.1.

Před požadavkem na změnu otáček bylo čekáno jednu sekundu na ustálení magnetického toku. Následně byl motor roztočen na 350 rad/s, s nulovým zátěžným momentem.

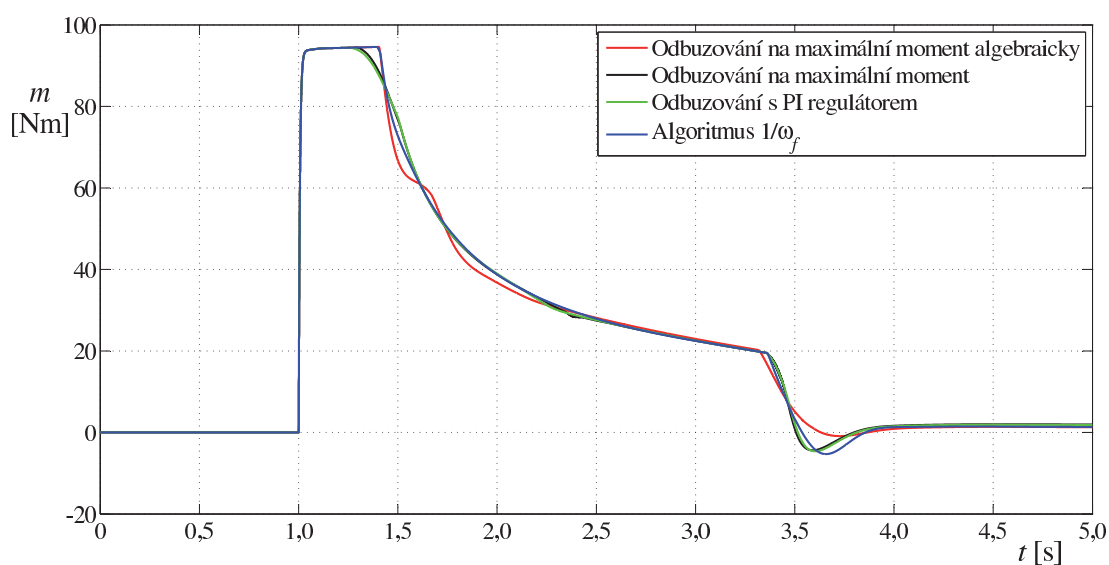
Stejně jako v kapitole 5.1, i zde byl u algoritmů  $1/\omega_f$  a na maximální moment algebraicky optimalizován parametr  $k_x$ . U algoritmu odbuzování s PI regulátorem a na maximální moment bylo  $k_x$  zvoleno a optimalizovány zesílení a integrační časové konstanty.

Veličina	Hodnota	Jednotka
$I_1^{max}$	80	A
$J$	0,3	kgm <sup>2</sup>
$k_{1/\omega f}$	0,892	-
$k_{Algeb}$	0,952	-
$k_{PI}$	0,950	-
$k_{Tmax}$	0,980	-
$L_1$	15,08	mH
$L_2$	15,08	mH
$L_h$	13,28	mH
$m_z$	0	Nm
$P_p$	2	-
$R_1$	0,41	$\Omega$
$R_2$	0,35	$\Omega$
$T_s$	120	$\mu$ s
$T_{s2}$	480	$\mu$ s
$T_{sim}$	12	$\mu$ s
$U_1^{max}$	200	V
$\sigma$	0,2245	-
$\Psi_{2d}^{max}$	0,5	Wb
$\Psi_{2d}^{min}$	0,2	Wb

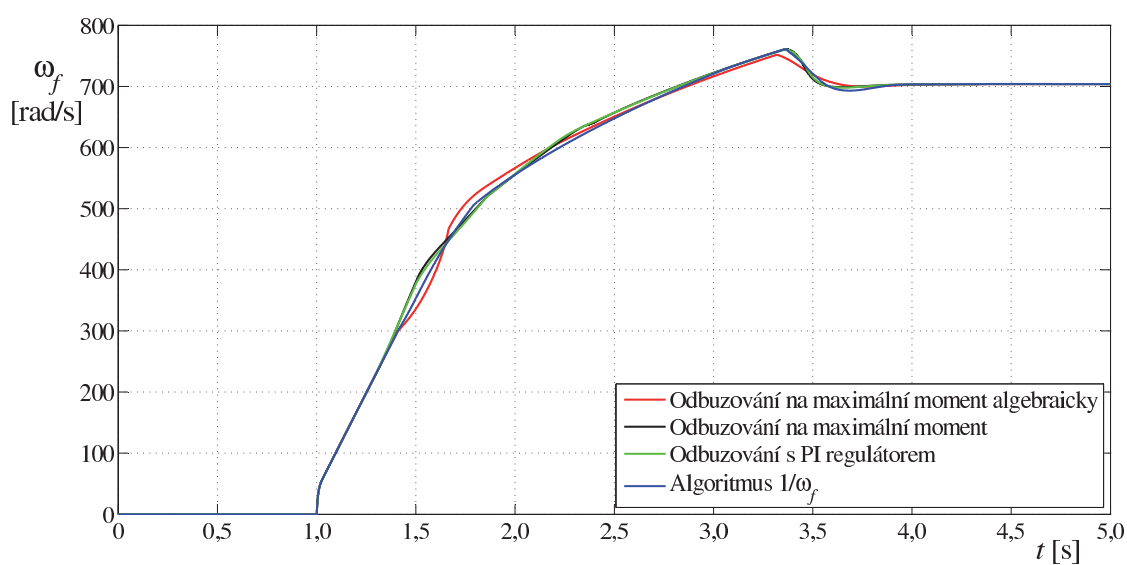
Tab. 3: Vstupní parametry simulace motoru s dosažitelným druhým regionem



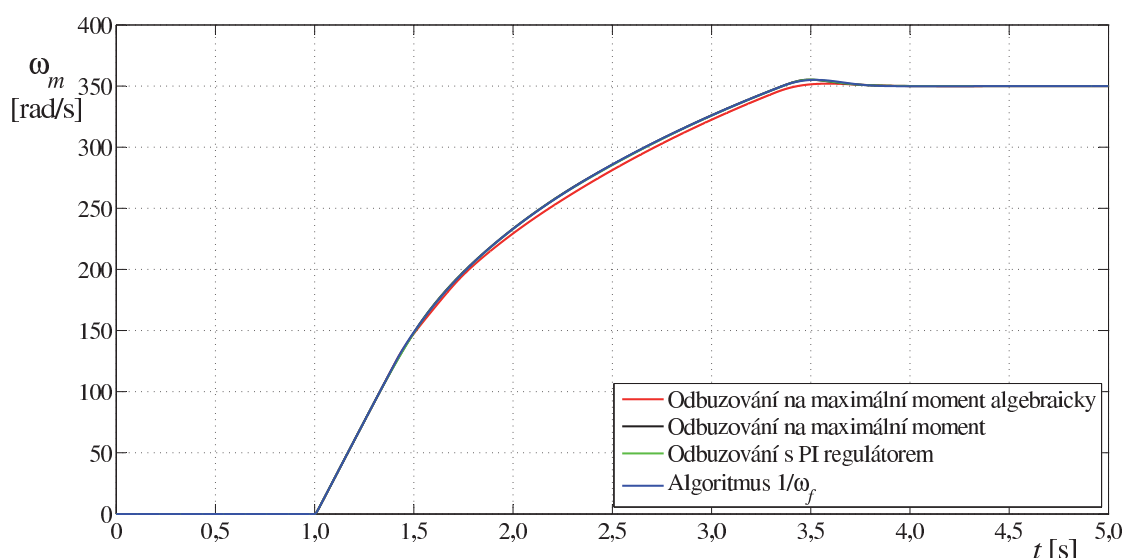
Obr. 23: Časová závislost magnetického toku u simulace motoru s dosažitelným druhým regionem



Obr. 24: Časová závislost momentu u simulace motoru s dosažitelným druhým regionem



Obr. 25: Časová závislost úhlové rychlosti  $\omega_f$  u simulace motoru s dosažitelným druhým regionem



**Obr. 26:** Časová závislost úhlové rychlosti rotoru u motoru s dosažitelným druhým regionem

Výsledky simulace jsou na obrázcích Obr. 23 až Obr. 26. Hodnoty ITAE jsou uvedeny v tabulce Tab. 4. Relativní srovnání  $\delta_{ITAE}$  je počítáno dle vztahu (5.1.1).

Patrný je především velmi špatný výsledek odbuzování na maximální moment algebraicky. Při bližším pohledu na časový průběh magnetického toku na obrázku Obr. 23 lze vidět, že magnetický tok je v druhém regionu nastavován příliš nízko, což vede ke ztrátě momentu. Lepších výsledků by šlo dosáhnout samostatnou optimalizací (zvýšením) reference magnetického toku a snížením rychlosti  $\omega_{base2}$ . Zanedbání, k nimž došlo v kapitole 3.4, však mají opačný charakter. Navíc vztahy, podle kterých je reference magnetického toku počítána, pak při přechodu mezi prvním a druhým regionem nenavazují a vzniká schod, jež se projeví v zákmitu momentu. Chyba se tedy pravděpodobně nachází v nepřesnosti vztahů (3.4.5) a (3.4.6). Ostatní algoritmy dosahují obdobných výsledků.

Algoritmus odbuzování	ITAE [rad]	$\delta_{ITAE}$ [%]
na maximální moment	191,41	0,00
$1/\omega_f$	191,92	0,27
s PI regulátorem	192,29	0,46
na maximální moment algebraicky	199,91	4,44

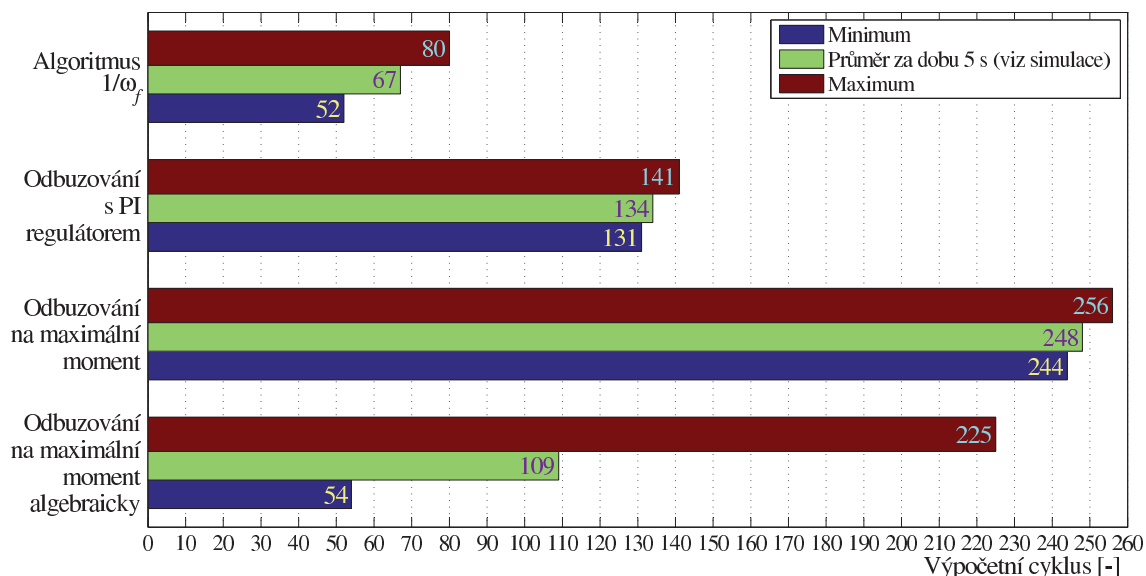
**Tab. 4:** Hodnoty ITAE kritéria dosažených algoritmy odbuzování při simulaci motoru s dosažitelným druhým regionem

### 5.3 Srovnání algoritmů odbuzování podle doby výpočtu

Doba výpočtu byla měřena pomocí příslušných funkcí obsažených v SFIO Toolboxu, na cílové straně.

Algoritmy odbuzování byly napsány jako `inline` funkce v jazyce C, s využitím intrinsických funkcí. Byla tedy zachována snadná čitelnost jejich kódu a hlavně bylo dosaženo zkrácení výpočetní doby.

Průměrné, minimální a maximální naměřené doby výpočtu (myšleno množství výpočetních cyklů) všech implementovaných algoritmů jsou v grafu na obrázku Obr. 27. Hodnoty byly získány měřením při simulaci motoru s dosažitelným druhým regionem po dobu 5 s (viz kapitole 5.2). Proměnná doba výpočtu je způsobena přítomností větvení. Jelikož řízení probíhá v reálném čase, je nutno vždy uvažovat nejhorší případ, tedy maximální dobu výpočtu.



Obr. 27: Srovnání algoritmů podle výpočetní doby

Z předchozího grafu tedy vyplývá, že algoritmy uvažující oba regiony (na maximální moment a na maximální moment algebraicky) jsou obecně náročnější na výpočet. Výpočetně nejméně náročným je algoritmus  $1/\omega_f$ .

Relativní srovnání maximální délky výpočtu algoritmů odbuzování  $k_{FWA}$  vůči maximální celkové době výpočtu algoritmu vektorového řízení  $k_{VOC}$ , včetně algoritmu odbuzování, je v tabulce Tab. 5.

Při zvolené vzorkovací periodě  $120\mu s$  a frekvenci procesoru 50MHz je výpočetní kapacita DSC 56F8346 nejvýše 6000 cyklů/ $T_s$ . Podle tabulky Tab. 5 má DSC 56F8346 dostatečnou rezervu výpočetního výkonu.

Algoritmus	$k_{\text{VOC}} [-]$	$k_{\text{FWA}} [-]$	$\delta_{\text{FWA}}^1 [\%]$
na maximální moment	1809	256	14,15
na maximální moment algebraicky	1779	225	12,65
s PI regulátorem	1779	141	7,93
$1/\omega_f$	1779	80	4,50

$$^1) \delta_{\text{FWA}} = 100 \cdot (k_{\text{FWA}} / k_{\text{VOC}})$$

**Tab. 5:** Výpočetní náročnost algoritmů odbuzování, vzhledem k celkové době výpočtu vektorového řízení

# ZÁVĚR

Úvodní část práce byla věnována modelování asynchronního motoru, teorii vektorového řízení a odbuzování při vysokých otáčkách. Na těchto základech byl následně vytvořen matematický model vektorového řízení asynchronního motoru v prostředí MATLAB-Simulink. Třetí kapitola se pak zabývala teorií vybraných algoritmů odbuzování. Těmi byly algoritmus  $1/\omega_f$ , odbuzování s PI regulátorem, odbuzování na maximální moment a odbuzování na maximální moment algebraicky.

Jedním z hlavních cílů práce byla příprava algoritmů odbuzování pro další testování na reálném systému. Tedy jejich implementace do DSC řady 56F8346 firmy Freescale s výpočty v pevné řádové čárce. Tomuto byla věnována kapitola 4. Značné usnadnění implementace přinesla metoda Processor-in-the-loop. Ta umožnila jednoduché a rychlé debugování a ověření algoritmů s pomocí simulace v prostředí MATLAB-Simulink. K její realizaci bylo využito SFIO Toolboxu vyvinutého na VUT FEKT ÚAMT.

Algoritmy odbuzování byly napsány v jazyce C za použití intrinsických funkcí. Tím bylo dosaženo jejich efektivního a přehledného kódu. Zároveň bylo dbáno na jednoduchost jejich použití. Všechny použité vztahy byly normalizovány v kapitole 4.3. Pro použití na jiném asynchronním motoru tak stačí přepočítat příslušné konstanty. Spolu s algoritmy odbuzování byla implementována i zbylá, řídicí část vektorového řízení. Normalizace všech příslušných vztahů se nachází v kapitole 4.2.

Pátá kapitola se pak zabývá ověřením vlastností algoritmů odbuzování. Díky zanedbáním, jež byla uvažována při jejich odvození, však bylo nutno napřed provést optimalizaci jejich nastavení. Vzhledem k množství nelinearit v oblasti odbuzování byla zvolena Nelder-Meadova metoda. V praxi by bylo nutno provést optimalizaci ručně. Následně byly provedeny simulace pro dva asynchronní motory, kdy u prvního motoru nebylo možno dosáhnout druhého regionu odbuzování. V kapitole 5.3 bylo rovněž provedeno srovnání výpočetní náročnosti algoritmů.

Z výsledků simulací vyplývá, že složitější algoritmy odbuzování nemusejí nutně přinášet o mnoho lepší výsledky. Ty jsou, díky optimalizaci, podobné u všech algoritmů odbuzování. Dosažené hodnoty *ITAE* kritéria se vždy lišily maximálně v řádech jednotek procent. Tedy krom odbuzování na maximální moment algebraicky, kdy v druhém regionu byla reference magnetického toku nastavována příliš nízko. To, a pravděpodobně značná citlivost na nepřesnost parametrů náhradního modelu asynchronního motoru, jej činí v praxi jen obtížně použitelným. Rovněž odbuzování na maximální moment, ač v druhém regionu dosahuje nejnižší hodnoty *ITAE* kritéria, má nevýhody. Jeho použití u motorů s dominantním prvním regionem je zbytečné, neboť je zde shodný jako algoritmus odbuzování s PI regulátorem. Další nevýhodou je

jeho relativně vysoká výpočetní náročnost a nutnost nastavit dva PI regulátory u nelineárního systému. Algoritmy  $1/\omega_f$  a odbuzování s PI regulátorem jsou výpočetně méně náročné a přitom dosahují výsledků srovnatelných jako u předchozích algoritmů.

Na práci lze navázat testováním algoritmů odbuzování při řízení reálného motoru, kde se mohou projevit jevy, jež simulace zanedbaly. Následný výzkum by pak mohl směřovat k ověřování citlivosti algoritmů odbuzování na nepřesnost parametrů matematického modelu asynchronního motoru. Přesné hodnoty jsou totiž málokdy známy.



# LITERATURA

- [1] CAHA, Zdeněk a Miroslav Černý. *Elektrické pohony*. 1. vyd. Praha: SNTL, 1990, 359 s. ISBN 80-03-00418-7.
- [2] ŠUBRT, Jaroslav. *Elektrické regulační pohony I*. 1. vyd. Brno: FEKT VUT v Brně, 1987, 180 s.
- [3] *Clarke & Park Transforms on the TMS320C2xx* [online]. Texas Instruments, 1997 [cit. 2012-05-23]. Dostupné z: <http://www.ti.com/lit/an/bpra048/bpra048.pdf>.
- [4] LEPKA, Jaroslav a Petr STEKL. *3-Phase AC Induction Motor Vector Control Using a 56F8300 Device* [online]. Freescale Semiconductor, 2005 [cit. 2012-05-23]. Dostupné z: <http://cache.freescale.com/files/product/doc/AN1930.pdf>
- [5] PIVOŇKA, Petr. *Číslicová řídicí technika*, Brno: FEKT VUT v Brně, 2003, 151 s.
- [6] VAS, Peter. *Sensorless vector and direct torque control*. New York: Oxford University Press, 1998, 729 s. ISBN 01-985-6465-1.
- [7] PETRUŠKA, Lubomír. *Simulace řídicích struktur elektromechanických systémů*. Diplomová práce. FEKT VUT v Brně, 2010, 90s.
- [8] SANG-HOON, Kim a Seung-Ki SUL. Maximum torque control of an induction machine in the field weakening region. In: *IEEE Transactions on industry applications*. 1995, s. 787-794. 31, 4.
- [9] SANG-HOON, Kim a Seung-Ki SUL. Voltage control strategy for maximum torque operation of an induction machine in the field-weakening region. In: *IEEE Transactions on industry applications*. 1997, s. 512–518. 44, 4.
- [10] *56F8346/56F8146 Data Sheet* [online]. Freescale Semiconductor, 2007 [cit. 2012-05-23]. Dostupné z: [www.freescale.com/files/dsp/doc/data\\_sheet/MC56F8346.pdf](http://www.freescale.com/files/dsp/doc/data_sheet/MC56F8346.pdf).
- [11] *DSP56800E and DSP56800EX Reference Manual* [online]. Freescale Semiconductor, 2011, [cit. 2012-05-23]. Dostupné z: [www.freescale.com/files/dsp/doc/ref\\_manual/DSP56800ERM.pdf](http://www.freescale.com/files/dsp/doc/ref_manual/DSP56800ERM.pdf).
- [12] *Motor Control Library-User Reference Manual* [online]. Freescale Semiconductor, 2011, [cit. 2012-05-23]. Dostupné z: [www.freescale.com/files/microcontrollers/doc/user\\_guide/56800E\\_MCLIB.pdf](http://www.freescale.com/files/microcontrollers/doc/user_guide/56800E_MCLIB.pdf).
- [13] BLAHA, Petr a Pavel Václavěk. *SFIO Toolbox For MC56F8xx MC56Fxxx and PPC processors User's Guide*. Freescale Semiconductor, 2008, [cit. 2012-05-23].
- [14] WON YOUNG, Yang a Wenwu CAO. *Applied numerical methods using MATLAB®*. John Wiley, 2005, 509 s. ISBN 978-0-471-70519-2.

- [15] BĚLOUŠEK, Radim. *Náhradní zapojení asynchronních strojů*. Diplomová práce. Brno: FEKT VUT v Brně, 2011, 86 s.

# SEZNAM PŘÍLOH

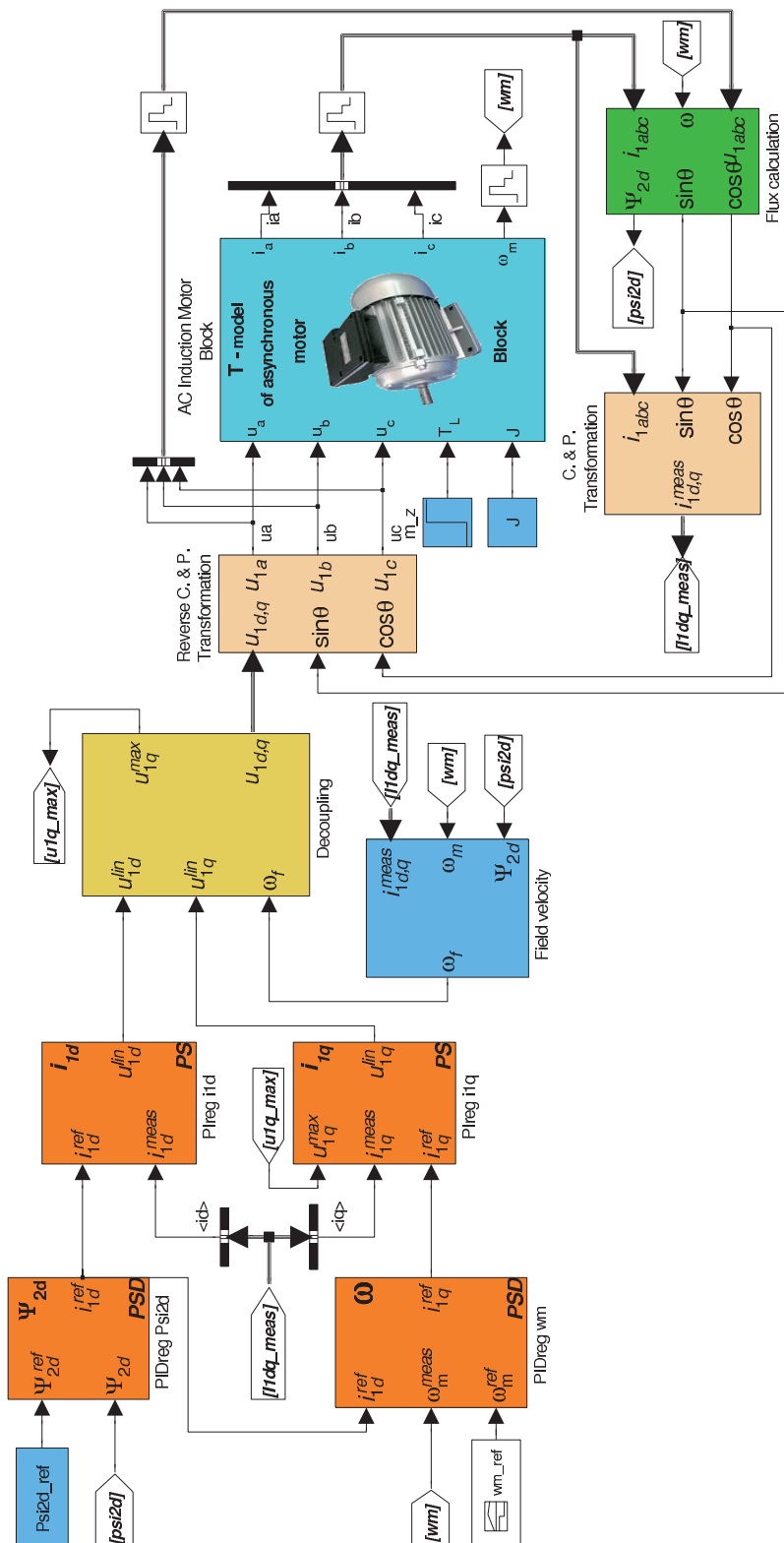
PŘÍLOHA 1: MATEMATICKÝ MODEL VEKTOROVÉHO ŘÍZENÍ  
V MATLAB-SIMULINK

PŘÍLOHA 2: VÝVOJOVÝ DIAGRAM VÝSLEDNÉHO ALGORITMU

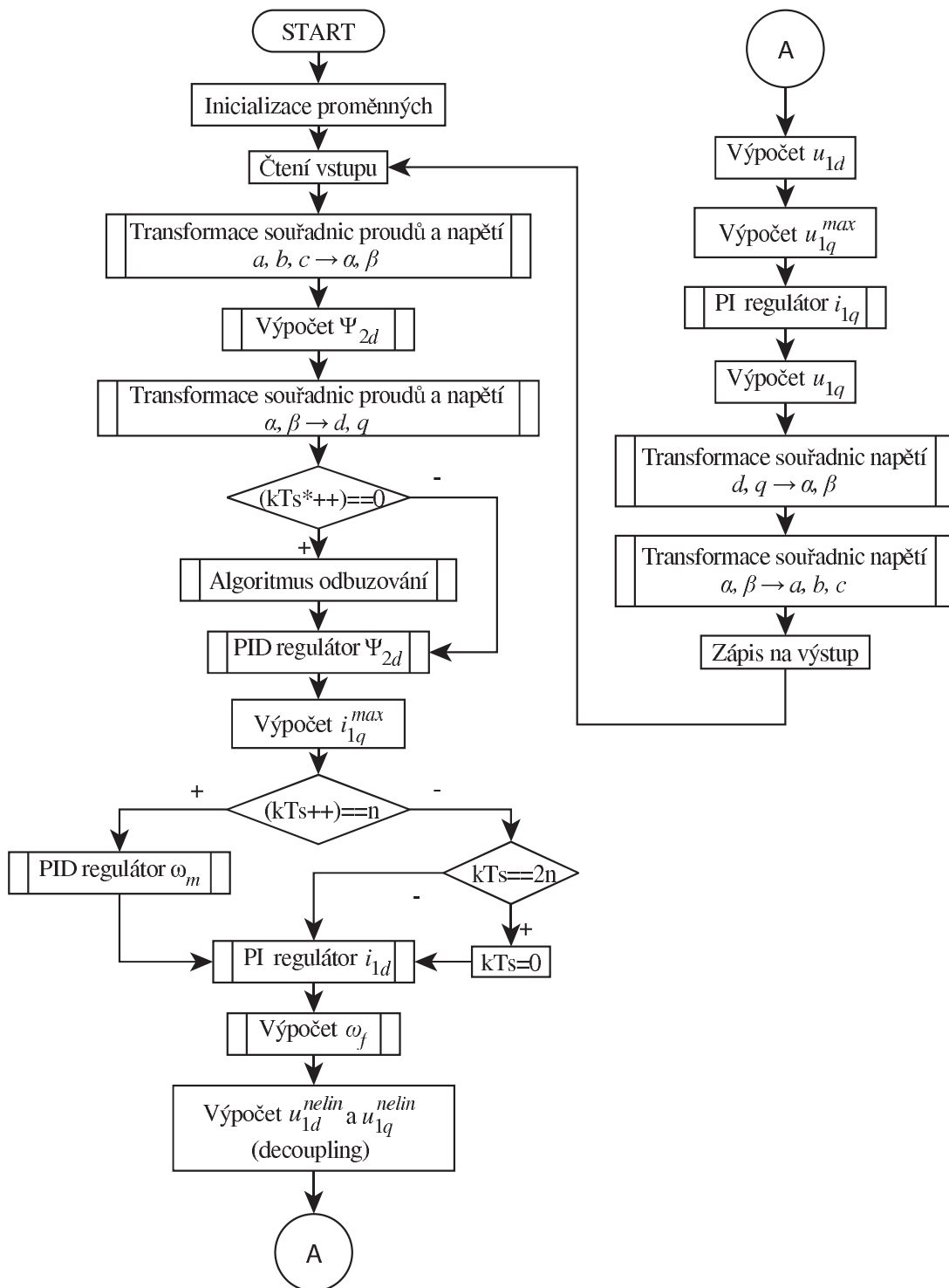
PŘÍLOHA 3: ZVOLENÉ HODNOTY NOREM

PŘÍLOHA 4: NASTAVENÍ REGULÁTORŮ

# MATEMATICKÝ MODEL VEKTOROVÉHO ŘÍZENÍ V MATLAB-SIMULINK



# VÝVOJOVÝ DIAGRAM VÝSLEDNÉHO ALGORITMU



\*  $kTs$  je pomocná proměnná, nesoucí pořadí vzorku

## ZVOLENÉ HODNOTY NOREM

Norma	Hodnota	Jednotka
$N_{i_l}$	11	A
$N_{\Psi_{2d}}$	1,2	Wb
$N_{u_l}$	250	V
$N_{\omega_f}$	500	rad/s
$N_{\omega_m}$	500	rad/s

**Zvolené hodnoty norem pro implementaci řídicích algoritmů do DSC 56F8346 u motoru s dominantní prvním regionem**

Norma	Hodnota	Jednotka
$N_{i_l}$	85	A
$N_{\Psi_{2d}}$	0,6	Wb
$N_{u_l}$	200	V
$N_{\omega_f}$	800	rad/s
$N_{\omega_m}$	400	rad/s

**Zvolené hodnoty norem pro implementaci řídicích algoritmů do DSC 56F8346 u motoru s dosažitelným druhým regionem**

# NASTAVENÍ REGULÁTORŮ

Umístění	Reg. veličina	Typ	Parametr					
			$K$ [-]	$T_i$ [ms]	$T_t$ [1/s]	$T_d$ [s]	$N$ [-]	$T_s$ [μs]
Vektorové řízení	$\Psi_{2d}$	PID	6,00	10,0	200	0,1	3	120
	$\omega_m$	PID	0,07	40,0	60	0,5	3	960
	$i_{1d}$	PI	139,20	173,0	200	-	-	120
	$i_{1q}$	PI	99,61	12,4	200	-	-	120
Odbuzování s PI regulátorem	$\Psi_{2d}^{ref}$	PI	$1,47 \cdot 10^{-6}$	5,8	200	-	-	960

Nastavení regulátorů pro simulaci asynchronního motoru s dominantním prvním regionem

Umístění	Reg. veličina	Typ	Parametr					
			$K$ [-]	$T_i$ [ms]	$T_t$ [1/s]	$T_d$ [s]	$N$ [-]	$T_s$ [μs]
Vektorové řízení	$\Psi_{2d}$	PID	7,00	3,5	200	0,5	3	120
	$\omega_m$	PID	1,50	50,0	100	1,0	3	480
	$i_{1d}$	PI	7,54	36,5	200	-	-	120
	$i_{1q}$	PI	0,56	8,2	200	-	-	120
Odbuzování s PI regulátorem	$\Psi_{2d}^{ref}$	PI	$10,07 \cdot 10^{-6}$	169,9	40	-	-	480
Odbuzování na maximální moment	$\Psi_{2d}^{ref}$	PI	$17,31 \cdot 10^{-6}$	138,1	40	-	-	480
	$i_{1q}^{max}$	PI	1,56	15,39	40	-	-	

Nastavení regulátorů pro simulaci asynchronního motoru s dosažitelným druhým regionem